Carlos Ivorra Castillo

LA LÓGICA DEL FINITISMO

El finitismo de Hilbert, con su requisito de ser "intuitivo", tiene una frontera muy poco natural.

KURT GÖDEL

Índice General

Introducción **vii**

Capítulo I: La aritmética recursiva primitiva **1**
1.1 El lenguaje de ARP 2
1.2 Demostraciones en ARP 23
1.3 Aritmética básica en ARP 34
1.4 El cálculo proposicional 44
1.5 Cuantificadores acotados 61

Capítulo II: Elementos de aritmética **69**
2.1 Más aritmética en ARP 71
2.2 Conjuntos finitos 87
2.3 Números enteros 101
2.4 Números racionales 106
2.5 Números primos 108

Capítulo III: La formalización de la lógica **115**
3.1 La formalización de ARP en ARP 115
3.2 Lenguajes formales de primer orden 131
3.3 La lógica de primer orden 148
3.4 Reglas derivadas de inferencia 158
3.5 Resultados adicionales 168
3.6 Teorías axiomáticas 177

Capítulo IV: Teorías aritméticas **185**
4.1 ARP como teoría de primer orden 185
4.2 La aritmética con inducción abierta 192
4.3 La jerarquía de Kleene 199
4.4 Aritmética básica en $I\Sigma_1$ 211
4.5 Funciones recursivas primitivas 217

Capítulo V: Teorías de conjuntos **227**
5.1 La teoría básica de conjuntos 227
5.2 Interpretaciones 240
5.3 La teoría de Zermelo 247

5.4 La teoría de Kripke-Platek 252
5.5 KP como teoría aritmética 259
5.6 Conjuntos finitos, cardinales 269
5.7 $I\Sigma_1$ como teoría de conjuntos 275

Capítulo VI: Teorías y metateorías **281**
6.1 La aritmética de Robinson 284
6.2 $I\Sigma_1$ y su formalización 290
6.3 Satisfacción de fórmulas de $\mathcal{L}_a$ 295
6.4 La Σ_1-completitud de Q 303
6.5 Teoremas de corrección 310
6.6 Teorías semirrecursivas 316

Capítulo VII: Lógica de segundo orden **321**
7.1 Teorías axiomáticas de segundo orden 322
7.2 La aritmética de segundo orden 329
7.3 El axioma de comprensión recursiva 336
7.4 El teorema de completitud semántica 349
7.5 Modelos no estándar 367
7.6 El axioma de comprensión aritmética 372
7.7 Recursión aritmética 380

Capítulo VIII: Teoría de la recursión **387**
8.1 Funciones recursivas 387
8.2 Caracterización aritmética 403
8.3 Funciones recursivas parciales 410
8.4 Funciones demostrablemente recursivas 418
8.5 Conjuntos recursivos 422

Capítulo IX: Incompletitud **429**
9.1 El primer teorema de incompletitud 430
9.2 El segundo teorema de incompletitud 437
9.3 El teorema de Tarski 442
9.4 Incompletitud y aritmética no estándar 443

Índice de Materias **454**

Introducción

La matemática formal Supongamos que un matemático nos dice que toda función continua $f : [a, b] \longrightarrow \mathbb{R}$ en un intervalo cerrado $[a, b]$ toma un valor máximo y un valor mínimo. Si le preguntamos a qué se refiere con "función continua", nos podrá explicar que es una función $f : [a, b] \longrightarrow \mathbb{R}$ que cumple cierta propiedad. Si le preguntamos qué es $\mathbb{R}$ y que es un intervalo $[a, b]$, también nos podrá explicar a qué se refiere con estos conceptos, probablemente introduciendo otros nuevos, como la relación de orden en $\mathbb{R}$, etc. Si le vamos preguntando qué significa cada uno de los conceptos de los que nos hable, al cabo de un número finito de preguntas lograremos que el matemático haya expresado todo su enunciado original en términos de dos únicos conceptos: "conjunto" y "pertenencia". Pero si le preguntamos a qué llama "conjunto" o qué quiere decir cuando dice que un conjunto pertenece a otro, ahí ya no podrá darnos ninguna definición que nos aclare a qué se refiere.[1] En su lugar nos dirá que no es posible definir esos dos conceptos, sino que ambos están determinados por los axiomas de la teoría de conjuntos. En efecto, la concepción moderna de la matemática es, en pocas palabras, la siguiente:

> Suponemos que existen unos objetos, a los que llamamos conjuntos, que no sabemos precisar lo que son, entre los cuales se da una relación de pertenencia (es decir, unos conjuntos pertenecen a otros), que tampoco sabemos explicar en qué consiste, pero suponemos que los conjuntos y la relación de pertenencia satisfacen unos axiomas, los axiomas de la teoría de conjuntos, que aceptamos sin justificación alguna, de modo que los teoremas matemáticos son las afirmaciones sobre los conjuntos que podemos demostrar lógicamente a partir de dichos axiomas.

Esta concepción de la matemática puede sorprender a alguien que haya estudiado algo de matemáticas, pero que no esté familiarizado con los tecnicismos

[1] Probablemente, lo que le gustaría respondernos es que un conjunto es una colección de conjuntos, y que cuando decimos que un conjunto x pertenece a otro y, estamos diciendo que x es uno de los integrantes de la colección de conjuntos que es y, pero si el matemático conoce bien su oficio sabrá que no puede respondernos esto sin estar engañándonos, porque si pretendemos definir un conjunto como una colección de conjuntos, eso nos obligaría a admitir que "la colección de todos los conjuntos" es un conjunto, pero razonando con "el conjunto de todos los conjuntos" se puede llegar a una contradicción conocida como la paradoja de Cantor.

relacionados con su fundamentación, porque viene a decir que los matemáticos se confiesan incapaces de explicar qué son los objetos de los que hablan y, a pesar de todo, se atreven a razonar sobre ellos. ¿Cómo es posible razonar sobre unos objetos supuestamente desconocidos?

Aquí es donde interviene la lógica matemática, que muestra cómo es posible razonar con rigor y precisión sin necesidad de saber nada sobre si las afirmaciones sobre las que razonamos son verdaderas o falsas, sin necesidad siquiera de atribuirles significado alguno.

Aunque en este libro tendremos ocasión de analizar con detalle a qué nos referimos cuando hablamos de razonar sin presuponer ningún significado a las afirmaciones involucradas, para hacernos una primera idea podemos pensar en este ejemplo:

Todo *orefimam* es un *odarbetrev,*
Todo *obol* es un *orefimam*,
luego Todo *obol* es un *odarbetrev.*

Esto es un ejemplo de razonamiento formalmente válido. No importa que las palabras involucradas no signifiquen nada. Si las dos premisas (las dos afirmaciones superiores) significaran algo y fueran verdaderas, la conclusión (la afirmación inferior) también significaría algo y sería verdadera. "Oficialmente" toda la matemática es así: consiste en partir de unos axiomas sobre conjuntos que no tienen por qué significar nada, pero de los cuales podemos deducir consecuencias formalmente igual que hemos deducido que todo *obol* es un *odarbetrev* sin necesidad de saber qué quiere decir eso. Si invertimos las palabras sin sentido obtenemos

Todo *mamífero* es un *vertebrado,*
Todo *lobo* es un *mamífero*,
luego Todo *lobo* es un *vertebrado.*

y podemos constatar que las premisas son verdaderas y que, consecuentemente, la conclusión también lo es, pero lo esencial es que no hacía falta asignar este u otro significado a las afirmaciones para asegurar que eran correctas, en virtud de un criterio puramente formal.

La lógica matemática permite explicitar todas las reglas posibles de razonamiento formal, similares a la que justifica que el razonamiento anterior es válido, de modo que razonar puede reducirse a aplicar reglas de inferencia formales sin atender para nada al posible significado de las afirmaciones consideradas.

Esto hace que el concepto de "teorema matemático", es decir, de afirmación matemática debidamente justificada, esté perfectamente definido sin necesidad de decir nada sobre qué son los conjuntos de los que se supone que hablan los teoremas matemáticos. Se podría incluso programar a un ordenador para que, si le damos una demostración matemática suficientemente detallada, el ordenador compruebe que la demostración es correcta sin necesidad de que entienda nada de álgebra, geometría, o la materia de la que trate la demostración. Sólo tendría que comprobar que cada afirmación en la demostración se puede deducir de las anteriores por alguna regla de inferencia formal avalada por la lógica matemática.

Formalismo Esta concepción formal de la matemática no ha sido adoptada por capricho, sino que es la única solución que han encontrado los matemáticos al problema con que se encuentran si intentan razonar aceptando cualquier cosa que parezca razonable aceptar sobre los conjuntos cuando se los define como "colecciones de objetos". Lo que se encuentran es que determinadas colecciones de objetos, como "el conjunto de todos los conjuntos" o "el conjunto de todos los conjuntos que no se pertenecen a sí mismos" dan lugar a contradicciones. Para más detalles sobre esto véase la introducción a mi libro *Lógica matemática* [LM]. Como explicamos allí, el hecho de que la concepción formal de la matemática se hubiera convertido en la salvación ante las contradicciones que surgían al tratar de razonar informalmente sobre conjuntos hizo que los matemáticos adoptaran como canon de rigor los requisitos de la lógica formal:

> Todo razonamiento matemático riguroso debe partir de axiomas explícitos de los que se extraen consecuencias según los principios de la lógica formal. Todos los conceptos involucrados tienen que estar determinados, o bien por dichos axiomas, o bien por definiciones que sólo aludan a otros conceptos previamente determinados (por los axiomas o por definiciones previas). En particular, no son admisibles definiciones o razonamientos basados en ideas o principios "intuitivos". Cualquier concepto o argumento "intuitivo" que quiera usarse en una demostración debe formalizarse debidamente. Cualquier definición o razonamiento "intuitivo" no formalizado es sospechoso de error e incluso de contradicción y, por lo tanto, es inadmisible. Es fácil poner una infinidad de ejemplos de cómo afirmaciones "intuitivamente" verdaderas resultan ser falsas a la luz de la matemática formal.

Esto es totalmente razonable si lo entendemos como los criterios que debe seguir un matemático al hacer su trabajo, pero muchos matemáticos acabaron siendo *formalistas,* en el sentido más fuerte de que consideraban que cualquier razonamiento matemático informal, en cualquier contexto, es cuestionable y, por lo tanto, carente de valor.

Aquí, por "razonamiento informal" queremos decir "razonamiento ordinario", es decir, el razonamiento que sí tiene en cuenta en todo momento lo que significan las afirmaciones consideradas cuyo criterio para aceptar cualquiera de ellas es que pueda justificar que es verdadera. El formalismo, entendido como el rechazo absoluto al razonamiento informal, no sólo es erróneo, sino que de hecho lleva a un callejón sin salida que impide a quien lo adopta entender la fundamentación de las matemáticas.

Decimos que el formalismo es erróneo porque una cosa es que sea exigible que toda afirmación matemática, para ser aceptada como teorema, deba acompañarse de una demostración puramente formal que cumpla los requisitos anteriores, y otra muy distinta que no sea posible demostrar informalmente afirmaciones matemáticas con el mismo rigor y la misma fiabilidad que proporciona una demostración formal. En efecto, si le explicamos a un niño de diez años que la suma de números naturales es conmutativa, y lo hacemos bien, es decir,

no le pedimos que se lo crea porque se lo decimos nosotros y no le vamos a engañar, sino que le hacemos ver que no puede ser de otra manera, sería absurdo decir que le estamos engañando por demostrárselo informalmente, sin apelar a ningunos axiomas sobre conjuntos.

Pero, aunque esto se pudiera discutir, y un formalista insistiera en que el niño no entiende realmente la situación desde el momento en que no conoce una demostración formal de la conmutatividad de la suma, en realidad no merece la pena discutirlo, porque, como vamos a mostrar a continuación, hay una razón de más peso por la cual el formalismo no se sostiene.

Metamatemática Lo que no tiene en cuenta un formalista es que los criterios de rigor que pretende exigir al razonamiento matemático no son inmediatos. Por el contrario, es necesaria una teoría lógica que construya lenguajes formales, es decir, lenguajes con una gramática que permita distinguir las cadenas de signos con sentido como $2 + 2 = 4$ de las que no tienen sentido, como $23 + + == 0$ por criterios puramente formales, que no aludan al posible significado de las cadenas con sentido. Es necesario definir y estudiar operaciones entre estas cadenas, como la que nos permite pasar de $x = 0$, $y = 0$, $xy = 0$ a

$$xy = 0 \to (x = 0 \vee y = 0).$$

Luego hay que determinar qué requisitos debe cumplir una sucesión de afirmaciones para que pueda considerarse un razonamiento formal válido, y todo esto requiere considerar, por ejemplo, números naturales y sus propiedades, por ejemplo, para llevar a cabo razonamientos por inducción sobre el número de signos de una afirmación, o sobre el número de afirmaciones de una deducción, etc.

La parte de la lógica que se ocupa de todo esto, es decir, de diseñar las teorías axiomáticas que permiten razonar formalmente y estudiar sus características, se conoce como *metamatemática.*

Ahora bien, si a un formalista le preguntamos qué son los números naturales, nos dirá que son unos conjuntos que se definen y cuyas propiedades se demuestran —formalmente, por supuesto— a partir de los axiomas de la teoría de conjuntos (u otra teoría apropiada). Pero si el formalista abre un libro de lógica matemática en el que se exponga la metamatemática necesaria para trabajar con teorías axiomáticas, se encontrará con que en él ya se habla de números naturales y se usan sus propiedades: los números naturales y sus propiedades se usan para determinar los criterios de razonamiento formal que permiten deducir formalmente de los axiomas de la teoría de conjuntos lo necesario para definir los números naturales y demostrar sus propiedades. ¡Un círculo vicioso en toda regla! La matemática formal necesita un fundamento lógico metamatemático y la metamatemática requiere conceptos —como los números naturales— que, para el formalista, deberían definirse y estudiarse mediante la matemática formal que la metamatemática pretende construir.

Finitismo El error fundamental del formalista está en creer que el razonamiento formal es el único refugio seguro ante los posibles errores y contradicciones en que se puede incurrir si se razona informalmente: el razonamiento formal

es la civilización, y fuera sólo está la jungla donde nada es seguro. En realidad hay que distinguir dos niveles de razonamiento matemático:

- Por una parte están los conceptos, afirmaciones y argumentos que podemos llamar *finitistas,* en cuanto que sólo involucran conjuntos finitos o, a lo sumo, sólo tratan con conjuntos infinitos de forma muy limitada.

  En este contexto nada impide razonar informalmente. Por ejemplo, para entender —informalmente— por qué $2 + 3 = 3 + 2$ basta observar esta figura:

  $$\bullet \quad \bullet \quad \circ \quad \circ \quad \circ$$

  y darse cuenta de que tenemos 5 puntos tanto si primero hemos dibujado los puntos negros y luego los blancos como si primero hemos dibujado los blancos y luego los negros. No va a haber más o menos puntos en función de si los contamos de izquierda a derecha o de derecha a izquierda. Esto es un razonamiento (informal) sobre cinco objetos en el que no hay margen de error posible. Y, más aún, podemos entender que el mismo razonamiento es aplicable en teoría sea cual sea el número de puntos negros y de puntos blancos que dibujemos, lo que nos asegura que $m+n = n+m$, para números naturales m y n cualesquiera. Con esto hemos llegado a una afirmación que involucra a los infinitos números naturales, pero se considera finitista, pues el argumento que nos convence de la igualdad $m + n = n + m$ para unos m y n dados sólo involucra $m + n$ objetos. Si a un niño de diez años le explicamos esto y lo entiende, podemos decir que ha entendido perfectamente (mediante un argumento informal) por qué la suma de números naturales es conmutativa.

- Por otro lado tenemos conceptos, afirmaciones y argumentos matemáticos que tratan esencialmente con conjuntos infinitos, pero, sobre todo, abstractos, a menudo más abstractos de lo que nos puede parecer que son.

  Por ejemplo, alguien se puede sorprender de que se pueda demostrar que existe una función $f : \mathbb{R} \longrightarrow \mathbb{R}$ que sea continua, pero no derivable en ningún punto. Uno podría sentirse tentado a concluir que es imposible que exista tal cosa, y si así lo hiciera, entraría en contradicción con la construcción formal que prueba que sí que existen tales funciones, pero ello se debe a que el concepto de función continua $f : \mathbb{R} \longrightarrow \mathbb{R}$ es mucho más abstracto y general que lo que pueda parecer a quien trate de identificar este concepto con el de una raya trazada sin levantar el lápiz del papel y sin pasar dos veces por la misma recta vertical. El concepto general de función continua $f : \mathbb{R} \longrightarrow \mathbb{R}$ no es finitista y no es de extrañar que nos equivoquemos si tratamos de razonar sobre él sin el marco de una teoría axiomática formal.

Para salir del círculo vicioso, el formalista necesitará persuadirse de que el razonamiento formal como garantía de rigor y de fiabilidad sólo es exigible a la hora de justificar afirmaciones no finitistas, mientras que las afirmaciones finitistas pueden justificarse informalmente con la misma fiabilidad que ofrece el razonamiento formal.

En estos términos podemos decir que la metamatemática requiere de razonamientos informales, lo cual es perfectamente admisible a condición de que éstos sean de naturaleza finitista. Con ellos podemos diseñar teorías axiomáticas formales en las que presentar formalmente tanto la matemática finitista —que no requiere realmente de tal presentación formalista— como la matemática no finitista, para la cual es indispensable el marco de una teoría axiomática formal, y el círculo vicioso desaparece.

Intuición El fundamento de los razonamientos formales lo proporciona la metamatemática, que establece los criterios precisos que determinan si un razonamiento formal es válido o no, mientras que los razonamientos informales finitistas se fundamentan en la *intuición,* donde usamos esta palabra en el sentido kantiano del término, y no en el sentido peyorativo que le da un formalista. Cuando hablamos de "intuición" no estamos hablando de corazonadas, de conjeturas, de un "yo sé que es así pero no sabría explicar por qué" ni de nada parecido. Intuición es la capacidad que tenemos de formarnos representaciones espaciales y temporales precisas. Por ejemplo, la figura con cinco puntos que hemos usado para justificar la conmutatividad de la suma es una representación intuitiva que hemos podido analizar para extraer consecuencias de ella igual que un jugador de ajedrez puede analizar una figura con una posición de una partida y encontrar la estrategia que lleva al mate en tres jugadas, o igual que un niño toma una figura que representa un laberinto y encuentra el camino que debe seguir el perro para llegar hasta el hueso. Todo son ejemplos de representaciones intuitivas que permiten extraer conclusiones objetivas totalmente fiables. Si el niño encuentra el camino del perro al hueso, ha demostrado que ese camino existe, cuando *a priori* podría no haber existido y, sin más que echar un vistazo siguiendo el camino marcado para comprobar que no ha hecho trampa pasando el lápiz por encima de ninguna pared, sería absurdo plantearse si la solución que ha encontrado el niño podría ser errónea porque no ha tenido en cuenta tal o cual exigencia de rigor formal.

En general, cualquier teorema matemático sobre grafos finitos, o sobre grupos finitos, o sobre combinatoria, etc. (si admite una demostración elemental, que no se apoye en conceptos auxiliares no finitistas) puede razonarse informalmente con el mismo rigor, objetividad y fiabilidad con que se razona la solución de un problema de ajedrez o se resuelve un *sudoku.* Cabría entonces preguntarse hasta dónde es posible razonar informalmente, qué resultados matemáticos pueden justificarse mediante razonamientos basados en la intuición, pero esto no nos preocupará —al menos directamente— en este libro, porque nuestro propósito es justo el contrario: ver cuáles son los mínimos hechos intuitivos necesarios para demostrar los resultados metamatemáticos que requiere la matemática formal (y veremos que son muy pocos).

La formalización de la metamatemática Aunque, en principio, el camino más simple para fundamentar la matemática es el que hemos descrito: desarrollar informalmente (mediante métodos finitistas) una metamatemática que nos dote de teorías axiomáticas formales en las que demostrar todos los resultados matemáticos, podemos convertir estos dos pasos en tres del modo siguiente:

1. Construimos y estudiamos informalmente una teoría axiomática formal muy simple.
2. En el seno de esta teoría formalizamos (es decir, presentamos razonando formalmente) los resultados metamatemáticos necesarios para construir y estudiar otras teorías axiomáticas, en particular las teorías de conjuntos en las que es posible demostrar formalmente todos los teoremas matemáticos.
3. Usamos estas teorías axiomáticas para demostrar todos los teoremas matemáticos.

Así, en lugar de razonar informalmente para construir la metamatemática que requieren las teorías axiomáticas formales, haremos esta construcción razonando formalmente en una teoría axiomática previa, que, inevitablemente, requerirá a su vez de una metamatemática informal. En realidad no trabajaremos con una única teoría básica, sino que estudiaremos varias, unas más potentes que otras. De este modo podremos discernir qué presupuestos requiere cada resultado metamatemático, según la teoría que necesitemos para poder demostrarlo formalmente.

Por supuesto, aunque un formalista resignado a admitir razonamientos metamatemáticos intuitivos pueda sentirse aliviado de que le mostremos cómo puede reducirlos al mínimo, esto no invalida que su formalismo sea teóricamente insostenible, y que, si quiere entender la fundamentación de la matemática, necesita admitir que es posible razonar informalmente sobre algunos conceptos intuitivos, especialmente sobre números naturales, y que no es coherente tratar de identificar éstos con ninguna construcción en el seno de una teoría axiomática formal. Más aún, veremos que la metamatemática proporciona argumentos específicos que prueban que ningún intento de definir formalmente los números naturales puede cumplir completamente su objetivo.

Otra razón por la que este enfoque es interesante es que es imposible demostrar que las teorías axiomáticas potentes que permiten demostrar todos los teoremas matemáticos son consistentes, es decir, que en ellas no es posible demostrar contradicciones. En cambio, las teorías que vamos a estudiar aquí no son teorías de conjuntos (diseñadas para que permitan hablar de conjuntos arbitrarios), sino teorías aritméticas (diseñadas para hablar, en principio, de números naturales, aunque son capaces de formalizar una porción relativamente amplia de las matemáticas) cuya consistencia puede ser demostrada por métodos finitistas.

Todas las teorías aritméticas que vamos a estudiar en este libro pueden definirse formalmente en la más simple de todas (de todas las que vamos a estudiar, aunque podríamos considerar algunas más simples aún), que es la conocida como *Aritmética Recursiva Primitiva* (ARP). Para ello empezaremos construyendo ARP informalmente (es decir, intuitivamente) y veremos que la propia construcción de ARP puede formalizarse en ARP, con lo que quedará claro que en la construcción de ARP no hemos hecho más supuestos intuitivos que los que están expresados por los (débiles) axiomas y reglas de inferencia de ARP. A continuación describimos brevemente las teorías formales más importantes que vamos a considerar, empezando por la que acabamos de mencionar:

ARP (La Aritmética Recursiva Primitiva) Es una teoría que permite hablar formalmente sobre los números naturales, pero únicamente permite definir propiedades "computables", es decir, tales que siempre podemos comprobar en la práctica si un número dado las satisface o no, y sólo podemos demostrar afirmaciones aritméticas que admiten pruebas constructivas. Así, en ARP no puede probarse que existe un número natural que tiene una determinada propiedad sin que se pueda mostrar explícitamente cómo calcularlo. En el capítulo I construiremos y estudiaremos ARP, mientras que en el capítulo II mostraremos cómo puede formalizarse en ARP la aritmética básica.

En el capítulo III formalizaremos en ARP la construcción de cálculos deductivos formales mucho más generales (lo que se conoce como teorías axiomáticas de primer orden) y en la sección 4.1 construiremos una teoría de primer orden a la que llamaremos también ARP y veremos que es esencialmente equivalente a la versión previa, en cuanto que permite formalizar algunas definiciones no "computables", pero no permite demostrar nada "sustancial" que no pueda probarse en la versión previa de ARP.

Podemos decir que ARP permite formalizar los razonamientos que se consideran "estrictamente finitistas", en el sentido de que no involucran más que conjuntos finitos.

AP (La Aritmética de Peano) Es una teoría mucho más potente que ARP, pues permite definir conceptos aritméticos "no computables" y formalizar algunos argumentos no constructivos. Sin embargo, todos sus axiomas pueden considerarse intuitivamente evidentes, por lo que todo razonamiento formal en AP se corresponde con un argumento intuitivo concluyente.

Introducimos AP en el capítulo IV junto con varias teorías más débiles, como IA (la aritmética con inducción abierta) y la que destacamos en el punto siguiente. En la sección 6.1 introduciremos también la Aritmética de Robinson Q, que es una teoría aún más débil que IA (es la aritmética de Peano sin el principio de inducción), pero que tiene interés porque algunos resultados generales sobre teorías aritméticas no requieren más hipótesis que la posibilidad de demostrar en ellas los axiomas de Q.

$\mathbf{I\Sigma_1}$ La teoría $I\Sigma_1$ resulta de restringir el principio de inducción la aritmética de Peano a una clase de propiedades especialmente simples, y su interés radica principalmente en que, aunque es ligeramente más potente que ARP, cualquier afirmación demostrable en $I\Sigma_1$ que sea formalizable en ARP es demostrable, de hecho, en ARP. Esto convierte a $I\Sigma_1$ en una formalización más práctica de la matemática estrictamente finitista.

B, $\mathbf{Z^*}$, KP En el capítulo V introduciremos varias teorías de conjuntos, empezando por la teoría básica B, de la que consideraremos diversas extensiones, como la teoría (restringida) de Zermelo Z^*, que es un fragmento de la teoría de conjuntos más habitual ZFC. Veremos que la aritmética de Peano AP es equivalente a la teoría $\mathrm{ZFC}_{\mathrm{fin}}$ que resulta de sustituir en

ZFC el axioma de infinitud (que postula la existencia de conjuntos infinitos) por su negación (que afirma que todos los conjuntos son finitos), en el sentido de que en AP y en $\mathrm{ZFC}_{\mathrm{fin}}$ se pueden demostrar exactamente los mismos teoremas aritméticos (y todo teorema de $\mathrm{ZFC}_{\mathrm{fin}}$ es demostrable en AP a través de cierto convenio para identificar los conjuntos finitos con números naturales).

La teoría de conjuntos de Kripke-Platek (KP) tiene con $\mathrm{I}\Sigma_1$ la misma relación que ZFC tiene con AP, es decir, si consideramos la teoría $\mathrm{KP}_{\mathrm{fin}}$ que incluye el axioma que postula que todo conjunto es finito, entonces los teoremas aritméticos de $\mathrm{KP}_{\mathrm{fin}}$ son exactamente los mismos que los de $\mathrm{I}\Sigma_1$, y todo teorema de $\mathrm{KP}_{\mathrm{fin}}$ puede probarse en $\mathrm{I}\Sigma_1$ identificando los conjuntos finitos con números naturales.

Podemos expresar esto diciendo que $\mathrm{ZFC} - \mathrm{AI}$ (es decir, ZFC sin el axioma de infinitud) y KP son teorías de conjuntos equivalentes a AP e $\mathrm{I}\Sigma_1$, respectivamente, salvo por el hecho de que dejan abierta la posibilidad de que existan conjuntos infinitos y son, por consiguiente, susceptibles de ser extendidas con un axioma de infinitud que introduzca tales conjuntos.

$\mathbf{ACR_0}$ La aritmética recursiva primitiva es suficiente para formalizar todos los resultados sobre teorías axiomáticas que son puramente sintácticos, es decir, que tratan exclusivamente sobre fórmulas (afirmaciones formales) y demostraciones. Sin embargo, ninguna de las teorías que hemos considerado es suficiente para estudiar los aspectos semánticos de la lógica matemática, es decir, los resultados que relacionan las fórmulas con su posible significado, y que son necesarios para determinar si una teoría axiomática formal es "razonable" en varios sentidos (como que podemos asegurar que sus teoremas serán verdaderos en la medida en que podamos asegurar que sus axiomas lo son), así como para justificar que determinadas fórmulas no son demostrables en determinadas teorías. La semántica de la lógica matemática descansa principalmente en el concepto de "modelo" de un lenguaje formal, y para formalizar este concepto es necesario trabajar con conjuntos infinitos más allá de lo que permiten las teorías que hemos considerado hasta ahora.

Una posibilidad es formalizar la teoría de modelos en cualquier teoría de conjuntos dotada de un axioma de infinitud, pero esto supone admitir que "todos los conjuntos" cumplen ciertos axiomas, y con ello perdemos todo el contacto con la matemática intuitiva, pues no podemos atribuir un sentido intuitivo concreto a afirmaciones sobre "la totalidad de los conjuntos".

En su lugar, en el capítulo VII introducimos la lógica de segundo orden, que a su vez nos permite definir teorías aritméticas de segundo orden, que hablan de números naturales y de conjuntos de números naturales. Aunque tampoco podamos dar sentido intuitivo a cualquier afirmación sobre "la totalidad de los conjuntos de números naturales", en el contexto restringido de la aritmética de segundo orden podemos cuidar de que los axiomas, o bien sean obviedades lógicas, o bien sean intuitivamente verdaderos.

El caso más sencillo es la teoría ACR_0, que sólo permite definir conjuntos de números naturales definibles mediante propiedades "computables" (a través del Axioma de Comprensión Recursiva). Veremos que ACR_0 es equivalente a $\mathrm{I}\Sigma_1$, en el sentido de que ambas teorías permiten demostrar los mismos teoremas sobre números naturales. En este sentido, ACR_0 es simplemente una forma más cómoda de formalizar la matemática estrictamente finitista. Los conjuntos definibles en ACR_0 son los mismos conjuntos de los que podemos hablar indirectamente en $\mathrm{I}\Sigma_1$ a través de las propiedades que los definen. (Por ejemplo, en $\mathrm{I}\Sigma_1$ podemos definir el concepto de "ser un número primo", y con ello podemos demostrar implícitamente propiedades sobre el conjunto de los números primos, aunque éste no sea definible explícitamente en la teoría. En cambio, en ACR_0 podemos hablar explícitamente del conjunto de todos los números primos, pero no podemos probar nada sobre él que no pudiéramos probar indirectamente en $\mathrm{I}\Sigma_1$.)

LKD_0 La teoría LKD_0 resulta de añadirle a ACR_0 un axioma de segundo orden conocido como el Lema de König débil. Aparte de que podemos considerarlo como intuitivamente evidente, veremos que permite demostrar los mismos resultados aritméticos que ACR_0 y, por lo tanto, que $\mathrm{I}\Sigma_1$, pero en cambio permite demostrar teoremas de segundo orden que no son demostrables en ACR_0. Uno de los más notables es el teorema de completitud semántica de Gödel, que es uno de los resultados fundamentales de la lógica matemática.

ACA_0 La teoría ACA_0 resulta de generalizar el Axioma de Comprensión Recursiva de ACR_0 hasta el Axioma de Comprensión Aritmética, que permite definir conjuntos de números naturales mediante cualquier propiedad definible en la aritmética de Peano. El resultado es que ACA_0 permite demostrar exactamente los mismos teoremas que AP sobre números naturales, por lo que es en realidad una mera reformulación de la aritmética de Peano. No obstante, la posibilidad de hablar explícitamente de conjuntos aritméticos (y no sólo de forma indirecta mediante las propiedades que los definen) permite probar muchos resultados que no son formalizables en AP. En particular, en ACA_0 es posible demostrar el llamado Lema de König, que es más general que el Lema de König Débil, con lo que todos los teoremas de LKD_0 lo son también de ACA_0.

RA_0 La teoría RA_0 resulta de añadir a ACA_0 el principio de Recursión Aritmética, que afirma que podemos definir una sucesión de conjuntos $\{X_n\}_{n=0}^{\infty}$ de números naturales de forma recurrente, de modo que X_n se define (mediante una propiedad aritmética, sin cuantificar sobre conjuntos) supuesto que los conjuntos $\{X_i\}_{i<n}$ están ya definidos. Es la teoría más potente que vamos a estudiar aquí, y esta recursión aritmética es necesaria para formalizar las construcciones más básicas de la teoría de modelos. En particular, es necesaria para demostrar que los números naturales, con las operaciones aritméticas usuales, forman un modelo de AP.

Estas teorías deberían proporcionar una respuesta satisfactoria (y tranquilizante) a la pregunta que más preocupa al formalista mentalizado de que no tiene más remedio que aceptar un cierto número de razonamientos intuitivos para construir su paraíso formal que le permite prescindir "oficialmente" de la intuición de una vez para siempre: ¿qué se necesita suponer realmente para construir y estudiar la matemática formal? Aunque la respuesta natural es que basta razonar intuitivamente sobre conceptos muy elementales que incuestionablemente tienen un significado intuitivo preciso y permiten razonar informalmente sobre ellos con todo rigor, si uno quiere aislar la pequeña parte de la matemática intuitiva que realmente hace falta para construir y estudiar la matemática formal, se encuentra con que la parte puramente sintáctica (la definición de las teorías axiomáticas formales) no requiere más que aceptar como verdaderos los axiomas y reglas de inferencia de ARP, mientras que la parte semántica sólo requiere aceptar que podemos definir conjuntos de números naturales por comprensión aritmética (es decir, considerar el conjunto de todos los números naturales que cumplen cualquier propiedad aritmética) así como que podemos definir recurrentemente sucesiones de conjuntos de números naturales (donde cada conjunto se define aritméticamente a partir de los ya definidos).

Hasta aquí sólo hemos pretendido dar al lector una visión general del objeto de este libro, mientras que los apartados siguientes los dedicamos a discutir con detalle los pocos hechos intuitivos sobre los números naturales que vamos a necesitar para construir ARP. Terminamos este apartado advirtiendo al lector que hay una serie de resultados clave que hemos mencionado y que no demostraremos en este libro, porque requieren usar el llamado cálculo secuencial de Gentzen, así que para su demostración remitiremos a nuestro libro sobre *Cálculo secuencial* [CS], que puede ser leído simultáneamente a éste.

Los números naturales Tal y como hemos explicado en los apartados precedentes, a pesar de las reacciones alérgicas que ello pueda causar en un finitista, lo cierto es que no hay ningún problema en razonar informalmente con todo rigor con números naturales, y demostrar informal a la par que rigurosamente que todo número natural mayor que 1 se descompone unívocamente en producto de primos salvo el orden, y que todo número natural es suma de cuatro cuadrados, etc., al igual que es viable razonar informalmente con números enteros o racionales (mientras que, si entramos en los números reales, algo se puede decir, pero ahí ya tenemos que cuidar mucho lo que decimos). Sin embargo, para regocijo del formalista, como ya hemos anunciado, el propósito de este libro es precisamente mostrar que, si bien —mal que le pese— tenemos todo ese territorio a nuestra disposición, en realidad podemos renunciar a casi todo él y construir y estudiar la matemática formal sin salirnos de una pequeña parcela del mundo de la matemática intuitiva. En este apartado vamos a exponer lo poco que necesitamos saber sobre los números naturales —como conceptos intuitivos— para fundamentar la matemática formal.

Ante todo, el lector debe asumir que los números naturales son lo que cualquier niño educado de seis años sabe que son los números naturales, teniendo en cuenta que un niño no sabe nada de axiomas, ni de definiciones formales, ni de

deducciones formales. Si hemos de resumirlo en pocas palabras, lo que tenemos es esto:

1. Existe un primer número natural, al que llamamos *cero.*
2. Cada número natural tiene asociado otro al que llamamos su *siguiente.*
3. Si vamos enumerando los números naturales empezando con el cero, luego el siguiente de cero, luego el siguiente del siguiente de cero, y así sucesivamente, cada número que añadimos a la sucesión es distinto de los precedentes, y los números naturales son únicamente los que van apareciendo en esa sucesión.

Naturalmente, esto se presta a muchas reflexiones filosóficas. Trataremos de responder a las preguntas que más probablemente pueda formularse el lector:

- Se puede observar que estamos diciendo que hay un número natural al que llamamos cero y otros a los que llamamos siguiente de cero, siguiente del siguiente de cero, etc., pero con ello sólo estamos diciendo cómo llamamos a los números naturales, pero no estamos diciendo lo que son.

  La respuesta es que no hay nada más que decir. Cuando alguien piensa en algo llamado cero, que va seguido de algo llamado siguiente de cero, etc., entendiendo que se cumple 3, está pensando en los números naturales. Y esto sigue siendo así si en lugar de llamar cero al cero decide llamarlo *zero,* y si al siguiente de cero decide llamarlo *the next number after zero,* etc. Las palabras concretas que usemos para nombrar los números naturales son irrelevantes. Los números naturales no son unas palabras en concreto, ni unos signos concretos que usemos para representarlos, como $0, 1, 2, \ldots$, del mismo modo que un punto geométrico que podemos imaginar no es la palabra "punto" ni importa si lo llamamos así o de otro modo.

  En la teoría de conjuntos formal, es frecuente definir los números naturales de modo que el cero es el conjunto vacío $0 = \varnothing$, y el siguiente de cero es el conjunto $1 = \{0\}$, y el siguiente del siguiente de cero es el conjunto $2 = \{0, 1\}$, etc., pero esto no es más que un convenio necesario para hablar de números naturales en una teoría en la que, por construcción, los únicos objetos que conoce son conjuntos. Afirmar que el siguiente del siguiente de cero es un conjunto al cual pertenecen el cero y el siguiente de cero no es una verdad profunda sobre los números naturales que sólo llegan a conocer los que estudian teoría de conjuntos, sino que es un convenio que, estrictamente considerado, es falso.

  Es como si, para hacer un sorteo con el que elegir a una persona de entre varias, asignamos a cada una un número y elegimos un papel al azar entre varios que contienen cada uno un número. Una cosa es que cada persona esté representada en el sorteo por un número y otra muy distinta creer que Juan es el número 3, aunque él no lo sepa.

  Del mismo modo, una cosa es que los números naturales estén representados por conjuntos en una teoría de conjuntos y otra creer que el 3 es un

conjunto aunque un niño de seis años no lo sepa. Igualmente sería ajeno al concepto de "número natural" pretender que el siguiente del siguiente de cero es la cadena de signos $SS0$, o que los números naturales están hechos de madera, etc. Los números naturales son meramente posiciones en una sucesión infinita con ciertas características (que empiece, avance a saltos y no termine nunca), sin que tenga sentido concretar de qué están hechos.

Por ello los tres puntos precedentes se pueden considerar a todos los efectos una definición intuitiva completamente rigurosa de lo que entendemos por "números naturales", en el sentido de que los números naturales no son ni más ni menos que lo que se entiende que son a partir de esos tres puntos. No falta nada y cualquier añadido sería algo ajeno al concepto de número natural.

- Un punto de la definición precedente que sin duda hará llevarse las manos a la cabeza a cualquier formalista que no tenga asimilado lo que supone dejar de serlo es el "y así sucesivamente" que aparece en el punto 3.

  En efecto, es posible que, en su infancia, el formalista viera con naturalidad ese "y así sucesivamente", pero en sus estudios le enseñaron que eso es inadmisible en un contexto formal riguroso, y en efecto es así: un "y así sucesivamente" no tiene cabida en una teoría axiomática formal, pero tiene pleno sentido cuando hablamos de conceptos intuitivos. Del mismo modo que cualquiera ve que un segmento de recta se puede prolongar indefinidamente sin que nunca deje de ser un segmento de recta cada vez más largo, el lector también entenderá que la sucesión:

  > *cero, siguiente de cero, siguiente del siguiente de cero, siguiente del siguiente del siguiente de cero, ...*

  se puede prolongar indefinidamente conservando siempre el mismo patrón de añadir un "siguiente" cada vez, por lo que el "y así sucesivamente" tiene un significado totalmente preciso. De hecho, demostraremos que ninguna definición formal puede capturar el sentido pleno de ese "y así sucesivamente".

- Se puede objetar que no tenemos una representación intuitiva del número "un millón", en el sentido de que si vemos un millón de puntitos dibujados en un papel no podemos decidir si estamos viendo un millón de puntitos o uno más o uno menos.

  Esto es cierto, pero nadie afirma que tengamos una representación intuitiva clara y distinta de cada número natural (pues no es así), del mismo modo que no podemos concebir intuitivamente un ángulo de 0.000000000001 grados sexagesimales como algo distinto de la imagen de un ángulo el doble de amplio. Lo que afirmamos es que tenemos una intuición clara de lo que supone ir prolongando indefinidamente una sucesión añadiendo un término adicional cada vez, y de las consecuencias que ello conlleva sobre las propiedades de los términos de dicha sucesión. No hay ninguna necesidad de concebir intuitivamente cada uno de sus términos.

Podemos razonar hechos generales sobre los números naturales mediante razonamientos en los que sea indudable que son aplicables a cualquier número natural sin necesidad de que podamos formarnos una imagen intuitiva de cada uno de ellos.

- Se puede filosofar sobre si los números naturales son una construcción de la mente humana o si tienen una existencia objetiva independiente de nuestras mentes. No necesitamos entrar aquí en esa polémica, pero sí puede ser relevante hacer algunas aclaraciones sobre ella. Si alguien considera que los números naturales son lo que son y que nosotros simplemente los estudiamos, sin poner ni quitar nada a lo que son, uno podría preguntarse cómo sabemos que, en efecto, un número natural es distinto de todos los anteriores. ¿Y si el siguiente del siguiente del siguiente de cero es el mismo cero y no lo sabemos?

  Eso no tiene sentido, porque los tres puntos que hemos usado como definición de los números naturales pueden verse indistintamente como nuestra "construcción" de los números naturales, de modo que podemos decir que cumplen eso porque los construimos para que cumplan eso, o también como una definición de las palabras "números naturales". Si uno piensa que los números naturales existen objetivamente, con independencia de la mente humana, tendrá que admitir que también existen otros conceptos similares, como el de una sucesión cíclica que se repite a partir de su quinto término, y por ello lo que hacemos al definir los números naturales es que, aunque, objetivamente con independencia de la mente humana, existe sucesiones que se ciclan y sucesiones que no se ciclan, reservamos la expresión "números naturales" para referirnos a una sucesión que no se cicla.

  Preguntarse cómo sabemos que los números naturales no se repiten es como preguntarse cómo sabemos que si pensamos en un decágono, el polígono en el que estamos pensando tiene diez lados y no nueve. Podemos pensar en un polígono de diez lados y en uno de nueve, como queramos, pero sólo decimos que estamos pensando en un decágono si pensamos en uno de diez lados. No podemos creer que estamos pensando en un polígono de diez lados y que en realidad estemos pensando en uno de nueve, y ello es compatible con el supuesto de que los eneágonos y los decágonos existan objetivamente, sin que nosotros podamos ponerles ni quitarles lados.

- Una cuestión menor es que el castellano tiene nombres para referirse a los números naturales, de modo que al siguiente de cero se le llama usualmente "uno", y al siguiente del siguiente de cero se le llama habitualmente "dos", etc. De hecho, en los apartados precedentes hemos hecho referencia a números como "diez", "nueve", "un millón", y hemos dado por hecho que el lector habrá entendido que con "diez" hacíamos referencia al siguiente del siguiente del siguiente del siguiente del siguiente del siguiente del siguiente del siguiente del siguiente del siguiente de cero.

  Más aún, el lector conoce sin duda el sistema de notación posicional que

nos permite nombrar a cualquier número natural con una sucesión de dígitos como 3 402. Esto presupone cierto conocimiento de la aritmética de los números naturales que podemos dejar fuera de la parcela de conocimiento intuitivo que necesitamos para construir la matemática formal. No obstante, dado que el lector conoce esta notación para representar los números naturales, sería una tortura mutua innecesaria que, para hacer referencia a un determinado número natural, en lugar de escribir 3 402, el lector nos obligara a escribir —o nosotros le obligáramos a leer— la expresión de la forma "el siguiente del siguiente del siguiente..." que lo nombra sin recurrir a la notación decimal.

Pero el formalista que desea reducir al mínimo el "gasto" necesario en intuición para fundamentar la matemática formal, cual avaro al que le duele cada céntimo gastado en su propio sustento, puede considerar con alivio que, cada vez que hablemos, por ejemplo, del número 11, no es necesario presuponer la aritmética que, en última instancia, se requiere para interpretar esta expresión.

Una vez discutida la definición intuitiva de número natural, vamos a destacar las pocas consecuencias que se desprenden de ella y que vamos a necesitar:

La primera es que los números naturales tienen un orden natural: si m y n son dos números naturales, puede suceder que m sea *menor que* n, y lo representaremos por $m < n$, que sea *mayor que* n y lo representaremos por $m > n$, o que ambos sean el mismo número natural, y lo representaremos por $m = n$.

La definición de este orden natural es muy simple: decimos que $m < n$ si m aparece antes que n en la sucesión:

cero, siguiente de cero, siguiente del siguiente de cero, siguiente del siguiente del siguiente de cero, ...

mientras que $m > n$ si m aparece después que n.

El formalista que tenga que hacer de tripas corazón para aceptar esta definición que contraviene todos los estándares de rigor aplicables en el contexto de la matemática formal debería esforzarse por entender que no hay contradicción en considerar inadmisible una definición así en un contexto en el que "antes" no significa nada y considerar totalmente precisa y objetiva esta definición en la que tenemos que partir de que sabemos perfectamente qué significa "antes". Razonar intuitivamente sobre los números naturales requiere hacer uso de nuestra intuición temporal, es decir, de nuestra capacidad de situar sucesos en el tiempo y distinguir entre sucesos anteriores y posteriores, pero aplicada a sucesos puramente intuitivos (no físicos).

Por ejemplo, de esta definición se desprende que, dados dos números naturales m, n, necesariamente se cumple uno (y sólo uno) de los casos $m < n$, $m = n$, $m > n$.

En efecto, sólo tenemos que ir enumerando: cero, siguiente de cero, siguiente del siguiente de cero ... y, por la definición de número natural que hemos dado, sabemos que tanto m como n aparecerán en ella. Si los dos aparecen a la vez es que son el mismo número natural (el que acaba de aparecer), y si uno aparece antes que el otro, ése es el menor. Naturalmente, aquí usamos lo que sabemos sobre el tiempo: dos sucesos cualesquiera tienen que estar ordenados en el tiempo: uno tiene que suceder antes que el otro, salvo que ambos sucedan a la vez.

No necesitamos aburrir al lector explicándole por qué si $m < n$ y $n < p$ entonces $m < p$, o cualquier otro hecho elemental sobre la ordenación de los números naturales. Usaremos la notación $m \leq n$ para indicar que m es menor o igual que n, e igualmente $m \geq n$ indicará que m es mayor o igual que n.

Mencionamos por último dos hechos fundamentales básicos sobre los números naturales:

Principio de buena ordenación *Si un número natural cumple una determinada propiedad, existe un mínimo número natural con dicha propiedad, es decir, un número natural que tiene la propiedad y ninguno menor la cumple también.*

Esto es inmediato: basta ir recorriendo la sucesión $0, S0, SS0, \ldots$ hasta que aparezca un número natural con la propiedad en cuestión. El primero que aparezca será el mínimo buscado.[2]

Principio de inducción *Si probamos que el número* 0 *tiene una propiedad y, suponiendo que un número arbitrario n la posee, podemos justificar que lo mismo le sucede a su siguiente, entonces podemos concluir que todos los números naturales tienen dicha propiedad.*

En efecto, si algún número natural no tuviera la propiedad, existiría un mínimo número natural sin ella. No podría ser 0, porque hemos probado que 0 la tiene, luego tendría que ser el siguiente de otro número natural n, pero como el siguiente de n es el menor número natural que no tiene la propiedad y n es menor que su siguiente, el número n debería tenerla, pero hemos razonado que en tal caso el siguiente de n también la tiene, con lo que tenemos una contradicción.

[2]El concepto de "propiedad" tiene la peculiaridad —desconocida en el contexto de la matemática formal— de que podemos identificar propiedades concretas bien definidas en el sentido de que sabemos lo que significa exactamente que un objeto (o un número natural en este contexto) la tenga, mientras que no poseemos una noción general de "propiedad" que nos permita hablar tranquilamente de "todas las propiedades". Por ello el principio de buena ordenación será aplicable siempre que podamos asegurar que la propiedad a la que queremos aplicarlo está bien definida, sin que podamos dar un criterio general de cuándo sucede tal cosa. De todos modos, para nuestro propósito sólo necesitaremos el principio de buena ordenación para propiedades que, no sólo estén bien definidas en el sentido de que sepamos qué significa que un número natural la cumpla, sino que siempre existirá un procedimiento explícito para determinar en un tiempo finito si cualquier número dado la cumple o no. Lo mismo se aplica al principio de inducción.

Aparte de estos hechos, sólo necesitamos suponer que el lector sabe contar, es decir, que si le preguntamos cuántos puntos hay aquí: •••••, sabe comprobar que hay cinco o, sin entrar en nomenclaturas particulares, que el número de puntos es el siguiente del siguiente del siguiente del siguiente del siguiente de cero.

Funciones recursivas primitivas Para construir y estudiar la matemática formal sólo necesitamos conocer los hechos básicos sobre números naturales que hemos discutido en el apartado anterior y el concepto de función recursiva primitiva que vamos a discutir seguidamente.

En general, si $n > 0$ es un número natural, una función n-ádica f (monádica, diádica, triádica, etc., según el valor de n) es cualquier criterio que a cada n objetos $a_1, \ldots, a_n$, repetidos o no y en un cierto orden, les asigna otro objeto, al que llamaremos su *imagen,* y que representaremos por $f(a_1, \ldots, a_n)$. En este apartado, cuando hablemos de funciones n-ádicas, entenderemos que nos referimos específicamente a funciones f que a cada n números naturales $a_1, \ldots, a_n$ (repetidos o no y en un cierto orden), les asigna otro número natural $f(a_1, \ldots, a_n)$.

Así, un ejemplo de función diádica es aquella para la que $f(m, n)$ es el máximo de m y n, de modo que, por ejemplo, $f(3, 5) = 5$, $f(7, 7) = 7$, etc. Es indudable que esta definición realmente determina una función diádica que podemos calcular en la práctica dado cualquier par de números naturales.

Se podría filosofar mucho sobre si una función está definida objetivamente aunque no sepamos cómo calcularla en la práctica, pero de momento podemos prescindir de tal discusión porque todas las funciones recursivas primitivas que vamos a introducir se pueden calcular explícitamente. Empezamos considerando unas funciones muy simples:

Funciones recursivas elementales Llamamos *funciones recursivas elementales* a las funciones siguientes:

- La función monádica S *(función sucesor)* que a cada número natural le asigna su siguiente.
- La función monádica c_0 *(función nula)* dada por $c_0(n) = 0$ para todo número natural n.
- Para cada par de números naturales $1 \leq k \leq n$, la función n-ádica p^n_k *(k-ésima proyección n-ádica)* dada por $p^n_k(m_1, \ldots, m_n) = m_k$.

Claramente, todas estas funciones pueden calcularse en la práctica sin dificultad. Más precisamente: es posible programar a un ordenador para que calcule cualquiera de ellas.

Ahora vamos a dar dos procedimientos para definir una nueva función a partir de otras dadas.

Composición Si tenemos definidas una función n-ádica H y n funciones m-ádicas $G_1, \ldots, G_n$, con ellas podemos formar una función m-ádica F dada por

$$F(x_1, \ldots, x_m) = H(G_1(x_1, \ldots, x_m), \ldots, G_n(x_1, \ldots, x_m)).$$

En otras palabras, para calcular $F(x_1, \ldots, x_m)$, calculamos todos los valores

$$G_1(x_1, \ldots, x_m), \ldots, G_n(x_1, \ldots, x_m)$$

y con ellos calculamos la función H, de modo que el resultado es, por definición, $F(x_1, \ldots, x_m)$.

Se dice que la función F está definida por *composición* a partir de las funciones $H, G_1, \ldots, G_n$.

Es inmediato que, dadas unas funciones $H, G_1, \ldots, G_n$ en las condiciones de esta definición, su composición está unívocamente determinada y, si un ordenador es capaz de calcular estas funciones para valores cualesquiera, también es capaz de calcular su composición.

Recursión Si G es una función n-ádica y H es una función $n+2$-ádica,[3] podemos definir como sigue una función $n+1$-ádica F determinada por las condiciones siguientes:

$$\begin{aligned} F(m_1, \ldots, m_n, 0) &= G(m_1, \ldots, m_n) \\ F(m_1, \ldots, m_n, k+1) &= H(m_1, \ldots, m_n, k, F(m_1, \ldots, m_n, k)). \end{aligned}$$

Se dice que F es la función definida por *recursión* a partir de G y H.

Para entender esto fijemos $n = 3$ para simplificar la notación, aunque esto es irrelevante. Si queremos calcular, por ejemplo, $F(1, 2, 3, 4)$, empezamos calculando

$$F(1, 2, 3, 0) = G(1, 2, 3).$$

Esto nos permite, a su vez, calcular

$$F(1, 2, 3, 1) = H(1, 2, 3, 0, F(1, 2, 3, 0)).$$

A su vez, con este valor ya calculado, podemos calcular

$$F(1, 2, 3, 2) = H(1, 2, 3, 1, F(1, 2, 3, 1)).$$

Similarmente se calcula $F(1, 2, 3, 3)$ y de ahí podemos calcular $F(1, 2, 3, 4)$.

[3] Escribimos $n+2$ en lugar de "el siguiente del siguiente de n" y un poco después $n+1$ en lugar de "el siguiente de n".

Es claro que cualquier par de funciones G y H (con n y $n+2$ argumentos, respectivamente) nos permiten calcular de este modo una función F. En otras palabras, para definir una función F, basta especificar cómo debe calcularse $F(m_1,\dots,m_n,0)$ (en términos de otra función G definida previamente) y cómo puede calcularse $F(m_1,\dots,m_n,k+1)$ supuesto que ya hayamos calculado $F(m_1,\dots,m_n,k)$ mediante una función H que use este valor junto el valor de k y los parámetros $m_1,\dots,m_n$.

También es claro que si un ordenador puede calcular las funciones G y H para valores cualesquiera, entonces también puede calcular la función definida por recursión a partir de ellas. Veamos algunos ejemplos concretos:

- Al componer la función S con la función p^3_3 obtenemos la función H dada por
$$H(m,n,r) = Sr.$$
- La función F definida por recursión a partir de p^1_1 y la función H anterior cumple
$$\begin{aligned} F(m,0) &= p^1_1(m) = m \\ F(m,n+1) &= H(m,n,F(m,n)) = S(F(m,n)) = F(m,n)+1. \end{aligned}$$
Hemos observado que hay una única función F que cumple estas dos propiedades, y si el lector sabe sumar, sabrá que la función $F(m,n) = m+n$ las cumple, pues, en efecto:
$$m+0 = m, \qquad m+(n+1) = (m+n)+1,$$
luego F no es más que la suma de números naturales. Si el lector prefiere prescindir del hecho de que sabe sumar, puede tomar la definición de F como definición de la suma de números naturales.
- La composición de la función suma con las funciones p^3_3 y p^3_1 nos da una función H^* determinada por
$$H^*(m,n,r) = F(p^3_3(m,n,r), p^3_1(m,n,r)) = r+m.$$
- La función F^* definida por recursión a partir de la función c_0 y de la función H^* anterior cumple
$$\begin{aligned} F^*(m,0) &= c_0(m) = 0 \\ F^*(m,n+1) &= H^*(m,n,F^*(m,n)) = F^*(m,n)+m. \end{aligned}$$
Nuevamente, existe una única función que cumple estas propiedades y, si el lector sabe multiplicar, sabrá que la función $F^*(m,n) = mn$ las cumple, pues
$$m\cdot 0 = 0, \qquad m(n+1) = mn+m.$$
luego la función F^* que hemos definido no es más que el producto usual de números naturales (y el lector puede tomar F^* como la definición de producto).

Ahora podemos definir las *funciones recursivas primitivas* como las funciones que pueden obtenerse a partir de las funciones recursivas elementales mediante composición y recursión.

Así, hemos probado que la suma y el producto usual de números naturales son dos ejemplos de funciones diádicas recursivas primitivas.[4]

A la hora de estudiar las funciones recursivas primitivas conviene dar una caracterización más práctica que la definición que hemos dado que concrete el "pueden obtenerse a partir de".

Para ello observamos que una función F es recursiva primitiva si y sólo si existe una sucesión de funciones $F_1, \ldots, F_r$ tal que F_r es la propia F y cada función F_i es recursiva elemental o bien se obtiene por composición o recursión a partir de funciones anteriores de la sucesión.

Por ejemplo, en el caso de la función producto, una sucesión que atestigua su carácter recursivo primitivo es:

$F_1(m,n,r) = r$	Recursiva elemental (p^3_3)
$F_2(r) = Sr$	Recursiva elemental (S)
$F_3(m,n,r) = Sr$	Composición de F_2 y F_1
$F_4(m) = m$	Recursiva elemental (p^1_1)
$F_5(m,n) = m + n$	Recursión con F_4 y F_3
$F_6(m,n,r) = m$	Recursiva elemental (p^3_1)
$F_7(m,n,r) = r + m$	Composición de F_5 con F_1 y F_6
$F_8(m) = 0$	Recursiva elemental (c_0)
$F_9(m,n) = mn$	Recursión con F_8 y F_7

Puesto que las funciones recursivas elementales pueden ser calculadas por un ordenador y las funciones definidas por composición o recursión a partir de funciones calculables por un ordenador son también calculables por un ordenador, concluimos claramente que todas las funciones recursivas primitivas son calculables por un ordenador, es decir, que, para cada función recursiva primitiva, existe un algoritmo que nos permite calcularla en un tiempo finito para números naturales cualesquiera (haciendo abstracción de las limitaciones de memoria que surgirían si los números fueran excesivamente grandes).

Podríamos dar muchos más ejemplos de funciones recursivas primitivas, pero en lugar de ello definiremos una teoría axiomática (la Aritmética Recursiva Primitiva) para hablar sobre ellas en un entorno "controlado" que nos permita estudiar la lógica subyacente.

Aquí termina lo que necesitamos saber para construir y estudiar la matemática formal: necesitamos saber lo que son intuitivamente los números naturales y que cumplen las propiedades básicas que hemos discutido, así como que las

[4] Las funciones recursivas primitivas fueron introducidas por Gödel, quien las llamó simplemente "funciones recursivas". Posteriormente, Herbrand introdujo una familia más general de funciones a las que llamó "recursivas generales", pero actualmente se llama funciones recursivas a las funciones recursivas generales de Herbrand y por ello las funciones recursivas de Gödel se llaman ahora funciones recursivas primitivas.

funciones recursivas elementales están bien definidas, y que mediante composición y recursión siempre podemos definir funciones nuevas a partir de otras ya dadas (cada una con el número de argumentos requerido), de modo que el concepto de "función recursiva primitiva" está bien definido y toda definición de una función recursiva primitiva define realmente una función que siempre puede ser calculada en la práctica en un tiempo finito. Eso es todo.

Conjuntos, relaciones y funciones Terminamos esta introducción precisando los conceptos intuitivos de "conjunto", "relación" y "función" porque los emplearemos ocasionalmente para motivar algunas definiciones, pero en realidad todos los apartados en los que hablamos de conjuntos, relaciones y funciones en general en términos intuitivos serán prescindibles, de modo que si el lector los omite podrá seguir igualmente el desarrollo de este libro.

Un conjunto M es una colección de objetos o, más precisamente, un criterio que nos permite afirmar si un objeto cualquiera está o no está en M. Esta definición es todo lo ambigua que es la palabra "criterio". En principio es posible definir conjuntos mediante criterios que no sepamos comprobar en la práctica si se cumplen o no, y no sabríamos dar condiciones explícitas que garanticen que un presunto "criterio" realmente determina un conjunto. Por ejemplo, "el conjunto de todos los conjuntos que no son elementos de sí mismos" es un concepto contradictorio, pues, si lo llamamos R, tenemos que si R es un elemento de sí mismo, no debería serlo, por la propia definición de R y, viceversa, si no es un elemento de sí mismo, debería serlo. No podemos explicar *a priori* por qué es contradictorio (lo es porque *a posteriori* vemos que de esa definición se sigue una contradicción). Pero, por otra parte, no debe extrañarnos que lo sea, porque R no tiene ningún contenido intuitivo. No tenemos la menor noción intuitiva de qué es "la totalidad de los conjuntos", como para extraer de ella los conjuntos que no se pertenecen a sí mismos. Podemos pensar en el conjunto R igual que podemos pensar en un espacio de 17 dimensiones, pero no tenemos ninguna imagen intuitiva de ninguna de las dos cosas, por lo que, aunque no supiéramos que R es contradictorio, sería inaceptable razonar informalmente sobre él.

Más aún, aunque nos restrinjamos a algo aparentemente mucho más simple, como sería el conjunto de todos los conjuntos de números naturales, éste sigue siendo algo en lo que podemos pensar, pero que no tiene ningún contenido intuitivo. Por ejemplo, cuando decimos que todos los números naturales cumplen una propiedad, queremos decir que 0 la tiene, y 1 la tiene, y 2 la tiene, etc., lo cual tiene sentido tanto si sabemos comprobar que así es como si no. En cambio, no podemos atribuir un significado intuitivo a una afirmación del tipo "todos los conjuntos de números naturales tienen tal propiedad".

Antes hemos razonado que todo conjunto de números naturales, si no es vacío, tiene un mínimo elemento. Esto tiene sentido porque disponemos de un argumento (informal) que nos asegura que, si tenemos un conjunto de números naturales bien definido y podemos asegurar que no es vacío, necesariamente tiene que tener un mínimo elemento, pero si hablamos de una propiedad tal que no disponemos de ningún argumento para probar que todos los conjuntos de números naturales la poseen, ni tampoco sabemos cómo mostrar un ejemplo de

conjunto que no la posee, no tenemos ningún criterio para afirmar que, o bien todos la poseen, o bien alguno no la posee, y con esto no estamos negando el principio lógico de que toda afirmación tiene que ser verdadera o falsa, sino que estamos afirmando que, en general, la afirmación "todo conjunto de números naturales tiene la propiedad tal" no tiene un significado preciso para que podamos afirmar que es verdadera o falsa.

Fijado un conjunto cualquiera M (lo cual supone que la pertenencia a M tiene un significado intuitivo inequívoco) y un número natural $n > 0$, una *relación* n-ádica R en M es cualquier criterio que puedan cumplir o no n elementos cualesquiera $a_1, \ldots, a_n$ de M repetidos o no y en un cierto orden. Si lo cumplen, lo representaremos con la notación $R(a_1, \ldots, a_n)$.

Por ejemplo, si llamamos $\mathbb{N}$ al conjunto de los números naturales, una relación diádica en $\mathbb{N}$ es la que se cumple sobre m, n cuando $m \leq n$, y la representaremos simplemente por $\leq$ (aunque escribiremos $m \leq n$ como es habitual en lugar de $\leq(m, n)$).

Como en el caso de los conjuntos, admitimos en principio que se pueda definir con precisión una relación en un conjunto sin que exista un criterio práctico para determinar si unos objetos la cumplen o no, y también tenemos que tener presente que el hecho de que podamos hablar de relaciones concretas bien definidas, no podemos atribuir ningún significado intuitivo *a priori* a una afirmación del tipo "toda una relación n-ádica en tal conjunto cumple tal propiedad" o "existe una relación n-ádica en tal conjunto que cumple tal propiedad" salvo que podamos razonar que así es.

Por último, una función n-ádica f de un conjunto M en otro conjunto N es cualquier criterio que a cada n objetos $a_1, \ldots, a_n$ de M (repetidos o no y en un cierto orden) les hace corresponder un objeto $f(a_1, \ldots, a_n)$ de N.

Las funciones recursivas primitivas son ejemplos de funciones (n-ádicas, para un n distinto en cada caso) de $\mathbb{N}$ en $\mathbb{N}$, y todas comparten la propiedad de que pueden calcularse en la práctica, pero en principio nada impide considerar funciones que estén bien definidas en el sentido de que sepamos exactamente a qué objeto nos referimos con $f(a_1, \ldots, a_n)$, pero que no sepamos cómo calcularlo en la práctica.

Por lo demás, todas las precauciones que hemos señalado sobre el uso informal de conjuntos o relaciones, especialmente lo tocante a hacer afirmaciones generales, se aplican también a las funciones.

Capítulo I

La aritmética recursiva primitiva

Tal y como explicábamos en la introducción, la matemática moderna más abstracta necesita presentarse como una teoría axiomática formal porque maneja conceptos tan abstractos que, por una parte, no tienen un contenido intuitivo preciso que nos permita razonar informalmente con ellos y, por otra parte, dan lugar a contradicciones si no se precisa cuidadosamente qué formas de razonamiento son admisibles.

Hilbert era consciente de que construir una teoría axiomática formal capaz de formalizar toda la matemática requería un trabajo previo informal, y consideró que era viable razonar informalmente sobre matemáticas a condición de que los razonamientos fueran estrictamente finitistas, es decir, que involucraran únicamente conjuntos y procesos finitos. En tal caso, el razonamiento intuitivo (riguroso, en el sentido de que todo cuanto se diga tenga un significado preciso y podamos asegurar que es cierto) es completamente fiable, pese al temor patológico de los formalistas.

Los límites del finitismo no son precisos, porque, por muy finitistas que seamos al tratar con números naturales, siempre tenemos que relacionarnos con conjuntos infinitos, como el propio conjunto de los números naturales. Aquí vamos a presentar una teoría axiomática que formaliza lo que podríamos considerar el finitismo más exigente. En realidad podríamos considerar teorías aún más restrictivas, pero sería difícil que alguien argumentara que la Aritmética Recursiva Primitiva (ARP) que vamos a estudiar aquí no es lo suficientemente finitista para que los razonamientos que permite formalizar pudieran ser sospechosos de esconder contradicciones, como le sucede a la teoría de conjuntos ingenua.

Veremos que se trata de una teoría lo suficientemente restrictiva como para que podamos asegurar que todo cuanto se demuestra en ella es intuitivamente verdadero, y a la vez suficientemente potente como para formalizar su propia construcción y también la construcción de ZFC o de cualquier teoría axiomática razonable. Por lo tanto, por una parte ARP es superflua en el sentido de que

todo lo que se demuestra en ella se puede demostrar informalmente con las mismas garantías y con menos tecnicismos, pero, por otra parte, nos servirá para delimitar los razonamientos intuitivos necesarios para construir y estudiar la matemática formal.

1.1 El lenguaje de ARP

En esta sección vamos a construir un ejemplo concreto de lo que se conoce como un lenguaje formal, es decir, un lenguaje dotado de una gramática que permite distinguir entre sinsentidos como "hoy si por casi no" y afirmaciones con sentido como "Hoy es jueves", pero estableciendo la distinción sin hacer referencia en ningún momento al posible significado de las presuntas afirmaciones consideradas, sino meramente a su forma (a su estructura gramatical). Pese a ello, lo cierto será que a cada afirmación del lenguaje que vamos a construir le podremos asignar un significado preciso, una afirmación sobre números naturales y funciones recursivas primitivas que podrá ser verdadera o falsa, pero será esencial que el lector constate que si decidiera no leer ningún párrafo que tenga en cuenta ese significado asignado a cada afirmación, ello no le impediría seguir los restantes sin que nada quedara sin justificar.

Definición 1.1 Llamaremos *signos del lenguaje* $\mathcal{L}_{\text{arp}}$ *de la Aritmética Recursiva Primitiva* a ocho signos distintos cualesquiera (es irrelevante cuáles elijamos) a los que llamaremos

$$0 \qquad S \qquad c \qquad p \qquad \kappa \qquad \rho \qquad x \qquad =$$

Esto puede entenderse de varias formas equivalentes. La más literal consiste en entender que, cuando hablamos de signos, nos referimos a eso, a cualquier trazo que podamos plasmar en un papel (o en una pantalla de ordenador) de forma que uno cualquiera de ellos se distinga inequívocamente de cualquier otro. (No sería buena idea, por ejemplo, considerar como signos distintos dos variantes de una p que se distingan por que el rabito de una es ligeramente mayor que el de la otra, ni otras elecciones que dieran lugar a problemas técnicos similares que sería absurdo enumerar aquí).

Pero otra interpretación más interesante es considerar que los signos de $\mathcal{L}_{\text{arp}}$ son conceptualmente como las palabras "rey blanco", "rey negro", "dama blanca", "dama negra", "torre blanca", "torre negra", etc. en una descripción del juego del ajedrez. Cuando hablamos de ajedrez, ¿qué es el rey blanco? Con esta expresión podemos referirnos a un objeto físico, una pieza de un juego de ajedrez, hecho de materia, pero también a un concepto abstracto que podemos usar en diagramas de tableros de ajedrez, o en listados de jugadas, como en R3CD (el rey (blanco) se mueve a la tercera casilla del caballo de la dama blanca), de modo que "rey blanco" pasa a ser un concepto igual de abstracto que el número "tres". El rey blanco no es nada en particular, sino un mero concepto del que podemos decir ciertas cosas determinadas por las reglas del ajedrez, igual que el número tres no es nada en particular, sino un mero concepto del que podemos decir ciertas cosas determinadas por la definición de los números naturales.

Aquí vamos a adoptar un convenio que es compatible con ambas interpretaciones, y es entender que, tal y como hemos dicho en la definición precedente, $0, S, c$, etc. NO son los signos de $\mathcal{L}_{\text{arp}}$, sino que son los nombres (en castellano) con los que nos referiremos a dichos signos.

El lector puede objetar que ρ no es ninguna palabra castellana, pero aquí tenemos que entender que ρ es una extensión del castellano análoga a la que consideramos cuando planteamos un caso diciendo que "A y B son dos hermanos que discuten sobre..." Igual que es lícito usar A y B para referirnos a los protagonistas de una historia, también podemos usar $0, S, c$, etc. para referirnos a los signos del lenguaje $\mathcal{L}_{\text{arp}}$, es decir, para nombrarlos, igual que usamos R, D, T, etc. para nombrar el rey, la dama, la torre, etc. de un juego de ajedrez, tanto si nos referimos a unas estatuillas concretas como a unos conceptos abstractos.

De este modo, un lector puede tomar como signo 0 de ARP el trazo "$\diamondsuit$", mientras que otro lector puede tomar como signo 0 el trazo "$\triangle$" y un tercer lector puede no elegir ningún trazo en particular, sino limitarse a pensar que 0 es un signo de $\mathcal{L}_{\text{arp}}$ igual que los peones son piezas del ajedrez, sin necesidad de darles ninguna forma concreta y representarlos por una P sin más. Así, cuando en este capítulo escribamos "0", el primer lector deberá entender que nos referimos a su signo "$\diamondsuit$", el segundo que nos referimos a su signo "$\triangle$" y el tercero que nos referimos al signo 0 de $\mathcal{L}_{\text{arp}}$, igual que alguien que está reproduciendo una partida de ajedrez en un tablero lee "$T3TD$" y entiende que la T hace referencia a una de las piezas de su propio tablero, una estatuilla que no conocía el que transcribió la partida que está leyendo.

Una *cadena de signos* de $\mathcal{L}_{\text{arp}}$ es cualquier sucesión finita de signos de $\mathcal{L}_{\text{arp}}$ con posibles repeticiones, pero en un orden prefijado.

En lo que sigue adoptaremos convenios diversos para nombrar distintas cadenas de signos de $\mathcal{L}_{\text{arp}}$, pero a falta de que introduzcamos otros criterios particulares, el criterio general que usaremos para nombrar cadenas de signos es el de nombrar sus signos en el mismo orden en que aparecen en la cadena.

Por ejemplo, $00S\kappa ==$ es una cadena de signos de $\mathcal{L}_{\text{arp}}$, que consta de seis signos, el primero de los cuales es 0, el segundo es otro 0, el tercero es S, etc., y no es la misma que $S0\kappa S ==$, que consta de los mismos signos, pero en otro orden.

Diremos que dos cadenas de signos ζ_1 y ζ_2 son *idénticas* (y lo representaremos por $\zeta_1 \equiv \zeta_2$, mientras que $\zeta_1 \not\equiv \zeta_2$ indicará que no lo son) si constan de los mismos signos dispuestos en el mismo orden.

Notemos que "$\zeta_1 \equiv \zeta_2$" es una abreviatura de una frase en castellano (y, como tal, la podemos considerar como una frase en castellano, o en una jerga técnica definida dentro del castellano) que expresa que "ζ_1" y "ζ_2" son nombres en castellano para una misma cadena de signos de $\mathcal{L}_{\text{arp}}$.

Si $\zeta_1, \ldots, \zeta_n$ son cadenas de signos de $\mathcal{L}_{\text{arp}}$, llamaremos $\zeta_1 \cdots \zeta_n$ a su *yuxtaposición,* es decir, a la cadena de signos que consta de los signos de ζ_1, seguidos de los signos de ζ_2, etc. hasta los signos de ζ_n.

Esto generaliza el convenio que hemos adoptado antes, por el que hemos acordado nombrar las cadenas de signos nombrando sus signos en el mismo orden.

A partir de aquí empezamos un proceso destinado a especificar qué cadenas de signos de $\mathcal{L}_{\text{arp}}$ serán significativas (es decir, tendrán un significado asociado) y cuáles no, pero lo esencial es que vamos a establecer la distinción sin hacer referencia en ningún momento a ningún posible significado de las cadenas de signos.

Numerales Si k es un número natural, llamaremos *numeral* de *índice* k a la cadena de signos N_k de $\mathcal{L}_{\text{arp}}$ formada por k signos S seguidos de un signo 0, de modo que

$$N_0 \equiv 0, \quad N_1 \equiv S0, \quad N_2 \equiv SS0, \quad N_3 \equiv SSS0, \quad N_4 \equiv SSSS0, \quad \ldots$$

Al signo 0 lo llamaremos *constante cero,* mientras que a S lo llamaremos *funtor sucesor.*

Como el lector ya habrá imaginado, el signo 0 será el nombre que daremos en $\mathcal{L}_{\text{arp}}$ al número natural 0, mientras que S nombrará a la función sucesor, con lo que cada numeral N_k será el nombre en $\mathcal{L}_{\text{arp}}$ del número natural k.

Pero recalcamos que decimos "será", "nombrará", etc., porque de momento no necesitamos asignarle ningún significado a los numerales. No necesitamos saber qué significado van a tener los numerales para reconocer si una cadena de signos dada es o no un numeral. Podemos programar a un ordenador para que, si le damos una cadena de signos de $\mathcal{L}_{\text{arp}}$, nos diga si es o no un numeral y cuál es su índice, sin necesidad de tener en cuenta que dicha cadena de signos nombrará en tal caso a un número natural.

En la práctica (pero no en teoría) podemos aprovechar que el lector conoce la notación posicional decimal para abreviar los numerales:

$$1 \equiv N_1 \equiv S0, \quad 2 \equiv N_2 \equiv SS0, \quad \ldots \quad 12 \equiv N_{12} \equiv SSSSSSSSSSS0, \quad \ldots$$

lo que sólo hay que entender como que, en lugar de escribir "$SSSSSSSSSSS0$", usaremos el nombre alternativo más breve "12" con la seguridad de que el lector sabrá a qué numeral nos estamos refiriendo, si bien en teoría siempre podríamos referirnos a él como $SSSSSSSSSSS0$. ∎

Nota Proponemos al lector una analogía que tal vez le ayude a asimilar convenientemente lo que estamos haciendo y, sobre todo, lo que vamos a hacer a continuación. Los antiguos alquimistas se referían a cierta sustancia como *"aqua uitae".* Por mucho que miremos la expresión *"aqua uitae",* no podremos saber de qué sustancia se trata si no lo sabemos ya de antemano, pero no ocurre lo mismo si miramos el nombre que le dan los químicos modernos a esa misma sustancia:

$$\mathrm{CH_3 - CH_2 - OH}$$

Esta fórmula es otra forma de referirse a ella, como lo es la expresión "alcohol etílico". Cualquiera de estas dos denominaciones revela la estructura química de la sustancia en cuestión, por lo que se trata de nombres que "se definen a sí mismos".

Análogamente podemos pensar que $SSS0$ es la "fórmula química" del número tres. Alguien que lea la palabra "tres" no puede saber de qué estamos hablando si no sabe de antemano que es el nombre en castellano del número tres, mientras que $SSS0$ es un nombre para el número tres que determina unívocamente el número al que nombra. La notación decimal posicional es otra "formulación química" alternativa más conveniente en la práctica, pero en teoría estamos usando otra conceptualmente más simple.

Por supuesto, para entender que $SSS0$ nombra al número tres es necesario saber que 0 nombra al número cero y que S nombra la operación siguiente, al igual que para entender la fórmula del alcohol etílico es necesario saber que C es carbono, H es hidrógeno, O es oxígeno, así como algunos principios de química, pero el caso es que, conociendo los nombres de los elementos y esos pocos principios, uno puede ver una fórmula química que no haya visto nunca antes y saber a qué sustancia hace referencia sin necesidad de consultar un "diccionario químico".

Al igual que hemos desarrollado una "formulación química" para los números naturales, de modo que cada numeral nos permite saber a qué número natural hace referencia, ahora vamos a introducir una "formulación química" para las funciones recursivas primitivas, es decir, vamos a definir unas "fórmulas" (que en este caso llamaremos "funtores") de modo que cada funtor nombre a una función recursiva primitiva, pero de tal modo que, viendo el funtor, podamos saber cuál es la función que nombra, igual que viendo una fórmula química podemos saber cuál es el compuesto que nombra. ∎

Como ya hemos indicado, usaremos el signo S como nombre para la función sucesor, mientras que el signo c de $\mathcal{L}_{\text{arp}}$ lo usaremos para nombrar la función que en la introducción hemos llamado c_0.

Las proyecciones Si $1 \leq k \leq n$ son números naturales, la cadena de signos $p_k^n \equiv pN_kN_n$ recibe el nombre de *proyección n-ádica k-ésima.* Por ejemplo:

$$p_2^3 \equiv pSS0SSS0$$

es una cadena formada por ocho signos de $\mathcal{L}_{\text{arp}}$. En cambio, las cadenas $p0SS0$ o $pSSS0SS0$ no son proyecciones.

Como el lector imaginará, usaremos estas cadenas de signos de $\mathcal{L}_{\text{arp}}$ para nombrar las funciones que en la introducción hemos llamado proyecciones, pero insistimos en que podemos programar a un ordenador para que determine si una cadena de signos es o no una proyección sin tener en cuenta para nada el significado que le daremos.

Funtores Pasamos ya a definir los funtores, que darán nombre a todas las funciones recursivas primitivas. Cada funtor tendrá asociado un número natural

no nulo al que llamaremos su *rango,* y a los funtores de rango n los llamaremos funtores n-ádicos (monádicos, diádicos, triádicos, etc.) Un primer esbozo de la definición sería el siguiente:

- El signo S es un funtor monádico, que nombra a la función sucesor.
- El signo c es un funtor monádico, que nombra a la función nula c_0.
- Cada proyección p_k^n es un funtor n-ádico que nombra a la proyección correspondiente.
- Si h es un funtor n-ádico que nombra a una función n-ádica H y $g_1, \ldots, g_n$ son funtores m-ádicos que nombran, respectivamente, las funciones m-ádicas $G_1, \ldots, G_n$, entonces[1]

  $$\kappa(h, g_1, \ldots, g_n) \equiv \kappa h g_1 \cdots g_n$$

  es un funtor m-ádico que nombra a la composición de las funciones H y $G_1, \ldots, G_n$.
- Si g es un funtor n-ádico que nombra la función n-ádica G y h es un funtor $n+2$-ádico que nombra la función $n+2$-ádica H, entonces

  $$\rho(g, h) \equiv \rho g h$$

  es un funtor $n+1$-ádico que nombra la función definida por recursión a partir de G y H.

Ejemplo Si llamamos

$$+ \quad \equiv \quad \rho(p_1^1, \kappa(S, p_3^3)) \equiv \rho p S 0 S 0 \kappa S p S S S 0 S S S 0,$$

tenemos que $+$ es un funtor diádico. Para comprobarlo observamos que podemos construirlo por pasos así:

f_1	S	funtor monádico
f_2	p_3^3	funtor triádico
f_3	$\kappa(f_1, f_2)$	funtor triádico
f_4	p_1^1	funtor monádico
f_5	$\rho(f_4, f_3)$	funtor diádico

y en la introducción hemos visto que $+ \equiv f_5$ nombra la suma usual de números naturales. ∎

[1] Estamos diciendo que usaremos la expresión "$\kappa(h, g_1, \ldots, g_n)$" para nombrar la cadena de signos $\kappa h g_1 \cdots g_n$, de modo que los paréntesis y las comas los introducimos únicamente para facilitar la lectura, pero la cadena $\kappa(h, g_1, \ldots, g_n)$ no contiene paréntesis ni comas (ni g's, ni subíndices) sino que consta únicamente de signos de $\mathcal{L}_{\text{arp}}$. Lo mismo se aplica a todas las definiciones en las que introducimos paréntesis u otros signos ajenos a $\mathcal{L}_{\text{arp}}$.

Nota Si el lector considera "ridículo" llamar algo tan simple como la suma con el nombre $\rho(p_1^1, \kappa(S, p_3^3))$ debe pararse a reflexionar sobre que nuestro objetivo no es cambiarle el nombre a la suma, sino desarrollar una "formulación química" capaz de nombrar cada función recursiva primitiva de modo que el nombre determine la función nombrada. La relación entre "+" y $\rho(p_1^1, \kappa(S, p_3^3))$ es la misma que hay entre "agua" y H_2O. No hay ningún inconveniente en seguir llamando "agua" al agua, e incluso sería una pedantería decir siempre "H_2O" en lugar de "agua". Del mismo modo, nosotros seguiremos llamando "+" a la suma, pero veremos que es utilísimo contar con una nomenclatura capaz de referirse a cada función recursiva primitiva de modo que el nombre determine la función, y en la nomenclatura que hemos construido, el nombre de la suma será el funtor $\rho(p_1^1, \kappa(S, p_3^3))$. ∎

El ejemplo anterior nos muestra la forma más operativa de definir con precisión el concepto de funtor, y destacamos que la definición siguiente es puramente formal, es decir, que en ella no hacemos referencia al significado que pretendemos darle a los funtores:

Definición 1.2 Una cadena de signos f de $\mathcal{L}_{\text{arp}}$ es un *funtor n-ádico* si existe una sucesión $f_1, \ldots, f_r$ de cadenas de signos y una sucesión de números naturales $n_1, \ldots, n_r$ de modo que $f \equiv f_r$ y $n = n_r$ y, para cada $i = 1, \ldots, r$, se cumple uno de los casos siguientes:

1. $f_i \equiv S$ y $n_i = 1$.
2. $f_i \equiv c$ y $n_i = 1$.
3. $f_i \equiv p_k^{n_i}$, para cierto $1 \leq k \leq n_i$.
4. $f_i \equiv \kappa(f_j, f_{i_1}, \ldots, f_{i_m}) \equiv \kappa f_j f_{i_1} \cdots f_{i_m}$, para ciertos $j, i_1, \ldots, i_m < i$, de modo que $n_j = m$ y $n_{i_1} = \cdots = n_{i_m} = n$.
5. $f_i \equiv \rho(f_j, f_k) \equiv \rho f_j f_k$, para ciertos $j, k < i$ de modo que $n_i = n_j + 1$, $n_k = n_j + 2$.

Esta definición expresa simplemente que una cadena de signos es un funtor si se puede construir a partir de los funtores elementales (S, c, p_k^n) aplicando las operaciones κ (composición) y ρ (recursión), cuidando de que los rangos de los funtores involucrados en cada paso sean los requeridos para que las composiciones y las recursiones tengan sentido.

Es claro que todas las cadenas f_i que aparecen en la definición anterior son funtores (pues satisfacen la definición con la sucesión $f_1, \ldots, f_i$), por lo que en la práctica, para definir y reconocer funtores, sólo necesitamos tener en cuenta que todo funtor f se encuentra necesariamente en uno de los casos siguientes:

1. $f \equiv S$ (funtor monádico).
2. $f \equiv c$ (funtor monádico).
3. $f \equiv p_k^n$, con $1 \leq k \leq n$ (funtor n-ádico).

4. $f \equiv \kappa(h, g_1, \ldots, g_n) \equiv \kappa h g_1 \cdots g_n$, donde h es un funtor n-ádico y $g_1, \ldots, g_n$ son funtores m-ádicos, y entonces f es un funtor m-ádico y se dice que *está definido por composición* a partir de los funtores $h, g_1, \ldots, g_n$.

5. $f \equiv \rho(g, h) \equiv \rho g h$, donde g es un funtor n-ádico y h es un funtor $n+2$-ádico, y entonces f es un funtor $n+1$-ádico y se dice que *está definido por recursión* a partir de los funtores g y h.

Ahora, *después* de haber definido formalmente los funtores, podemos dotarlos de significado. A cada funtor n-ádico f le asignamos una función n-ádica $F(f)$ con el criterio siguiente:

1. $F(S)$ es la función sucesor.
2. $F(c)$ es la función c_0 que toma siempre el valor 0.
3. $F(p^n_k)$ es la proyección n-ádica k-ésima.
4. $F(\kappa(h, g_1, \ldots, g_m))$ es la composición de $F(h)$ y $F(g_1), \ldots, F(g_m)$.
5. $F(\rho(g, h))$ es la función definida por recursión a partir de $F(g)$ y $F(h)$.

Ejemplos Cada vez que definimos un funtor en $\mathcal{L}_{\text{arp}}$, estamos definiendo una función recursiva primitiva. Veamos algunos casos concretos:

1. Consideremos los funtores

   $$c_0 \equiv c, \quad c_1 \equiv \kappa(S, c_0), \quad c_2 \equiv \kappa(S, c_1), \quad c_3 \equiv \kappa(S, c_2),$$

   y, en general, $c_{k+1} = \kappa(S, c_k)$. Alternativamente, c_k es el funtor que resulta de componer k veces con el funtor S el funtor c.

   En la práctica, cuando demos un nombre particular a un funtor f de $\mathcal{L}_{\text{arp}}$, usaremos el mismo nombre para nombrar la función $F(f)$, como ya hacemos con la suma, por ejemplo. Así, en el caso de los funtores c_k tenemos que sus funciones asociadas vienen dadas por

   $$c_1(n) = S(c_0(n)) = S(0) = 1, \quad c_2(n) = S(c_1(n)) = S(1) = 2,$$

   y, en general, la función c_k es la función constante que toma siempre el valor k. Vemos así que las funciones constantes son todas recursivas primitivas, y tenemos nombres para todas ellas.

2. Definimos $\text{pre} \equiv \kappa(\rho(c, p^3_2), c, p^1_1)$. Vamos a describir la función que nombra este funtor. Para ello, si llamamos $f \equiv \rho(c, p^3_2)$, tenemos que

   $$F(f)(m, 0) = F(c)(m) = 0, \quad F(f)(m, Sn) = p^3_2(m, n, F(f)(n)) = n.$$

   Por consiguiente, $\text{pre}(0) = F(f)(c(0), p^1_1(0)) = F(f)(0, 0) = 0$,

   $$\text{pre}(Sn) = F(f)(c(n), p^1_1(Sn)) = F(f)(0, Sn) = n,$$

   luego, en definitiva,

   $$\text{pre}(n) = \begin{cases} 0 & \text{si } n = 0, \\ n - 1 & \text{si } n > 0. \end{cases}$$

3. Ahora definimos $\dot{-} \equiv \rho(p_1^1, \kappa(\text{pre}, p_3^3))$. Así, si representamos

$$m \dot{-} n = F(\dot{-})(m, n),$$

tenemos que

$$m \dot{-} 0 = p_1^1(m) = m,$$
$$m \dot{-} Sn = \text{pre}(p_3^3(m, n, m \dot{-} n)) = \text{pre}(m \dot{-} n).$$

Así, cada vez que aumentamos en una unidad el segundo argumento, $m \dot{-} n$ es una unidad menor (hasta que llega a 0), por lo que $m \dot{-} n$ es la *resta truncada,* dada por:

$$m \dot{-} n = \begin{cases} 0 & \text{si } m \leq n, \\ m - n & \text{si } n \leq m, \end{cases}$$

pues esta función cumple las dos propiedades anteriores, y sólo puede cumplirlas una función.

4. A su vez podemos definir máx $\equiv \kappa(+, p_2^2, \dot{-})$, de modo que

$$\text{máx}(m, n) = p_2^2(m, n) + (m \dot{-} n) = n + (m \dot{-} n),$$

que claramente es el mayor de los dos argumentos m y n.

5. Similarmente, el funtor mín $\equiv \kappa(\dot{-}, p_1^2, \dot{-})$ nombra a la función diádica que calcula el mínimo de sus dos argumentos.

Vemos así como, a través de composiciones y recursiones sucesivas, podemos ir definiendo una amplia gama de funtores que nombran a otras tantas funciones recursivas primitivas. ∎

Continuamos nuestro análisis de los funtores de $\mathcal{L}_{\text{arp}}$ discutiendo algunos aspectos técnicos. En primer lugar, la definición de la función $F(f)$ asociada a un funtor puede explicitarse así:

Definición 1.3 Diremos que un funtor n-ádico f *denota* una función n-ádica F si existe una sucesión de funtores $f_1, \ldots, f_r$ en las condiciones de la definición 1.2 y una sucesión de funciones $F_1, \ldots, F_r$ de modo que F sea F_r y, para todo índice i, se cumpla:

1. Si $f_i \equiv S$, entonces F_i es la función sucesor.
2. Si $f_i \equiv c$, entonces F_i es la función c_0 que vale siempre 0.
3. Si $f_i \equiv p_k^n$, entonces F_i es la proyección correspondiente.
4. Si $f_i \equiv \kappa(f_j, f_{i_1}, \ldots, f_{i_m})$, entonces F_i es la composición de las funciones $F_j, F_{i_1}, \ldots, F_{i_m}$.
5. Si $f_i \equiv \rho(f_j, f_k)$, entonces F_i es la función definida por recursión a partir de F_j y F_k.

Es fácil concluir que cada funtor n-ádico f denota una única función F, que además es n-ádica y recursiva primitiva. En efecto, es obvio que toda sucesión en las condiciones de la definición 1.2 se puede completar hasta una sucesión de funciones en las condiciones de la definición anterior, por lo que todo funtor denota al menos una función, que claramente es recursiva primitiva. Para probar la unicidad suponemos que hay un funtor que denota dos funciones distintas (es decir, que toman valores distintos sobre unos mismos números naturales). Distinguimos varios casos:

- Es imposible que $f \equiv S$, $f \equiv c$ o $f \equiv p^n_k$, pues la definición de denotación sólo asigna una función posible a cada uno de estos funtores.
- Si $f \equiv \kappa(h, g_1, \ldots, g_n)$, alguno de los funtores h o g_i debe denotar dos funciones distintas, porque si cada uno de ellos sólo denotara una función, entonces la función denotada por f sería la composición de estas funciones, y también sería única.
- Si $f \equiv \rho(g, h)$, concluimos del mismo modo que alguna de las funciones g o h debe denotar al menos dos funciones distintas.

Así pues, si f denota dos funciones distintas, tiene que haber un funtor f_1 de menor longitud que también denote dos funciones distintas, y por la misma razón debería haber otro funtor f_2 de menor longitud que también denotara dos funciones distintas, y en definitiva tendríamos una sucesión infinita estrictamente decreciente de números naturales (las longitudes de los funtores que estamos obteniendo) y eso es imposible (ya que en tal caso no existiría el mínimo de las longitudes de los funtores de la sucesión, en contra del principio del mínimo).

Por consiguiente, dado un funtor n-ádico f, podemos definir *la función denotada por* f como la única función $F(f)$ denotada por f, que es una función n-ádica recursiva primitiva. Obviamente se trata de la misma función que ya habíamos definido recurrentemente, sólo que ahora tenemos una definición explícita.

Teniendo en cuenta la definición de función recursiva primitiva que hemos dado en la introducción, es inmediato que toda función recursiva primitiva es denotada por un funtor.

Así tenemos perfectamente distinguidos los funtores de $\mathcal{L}_{\text{arp}}$, que son cadenas de signos como $+ \equiv \rho(p^1_1, \kappa(S, p^3_3))$, de las funciones como $F(+)$, que es la suma de números naturales, es decir, la función que a cada par de números les asigna su suma, sin perjuicio de que, en la práctica, podamos usar el mismo nombre para representar un funtor y la función que denota.

Nota Estamos nombrando los funtores definidos por composición o recursión con expresiones de la forma "$\kappa(h, g_1, \ldots, g_m)$" y "$\rho(g, h)$", donde usamos paréntesis y comas para facilitar la lectura, pero $\mathcal{L}_{\text{arp}}$ no tiene paréntesis ni comas entre sus signos.

Esto es admisible porque, aunque la ausencia de paréntesis y comas puede complicar la lectura, no vuelve ilegibles los funtores, ni siquiera ambiguos. Vamos a verlo con un ejemplo. Consideremos la cadena:

$$f \equiv \rho c \kappa \rho p S0S0 \kappa SpSSS0SSS0pSSS0SSS0pS0SSS0$$

y vamos a ver que es un funtor. Más aún, vamos a ver que podemos interpretar esta cadena leyendo sus signos siempre de izquierda a derecha. El argumento es el siguiente:

1. Como f empieza por ρ, si es un funtor, tiene que estar definido por recursión. Por lo tanto, tras el signo ρ tienen que estar yuxtapuestos los nombres de dos funtores. El segundo signo es ya el funtor c (ningún otro funtor puede empezar por c), luego tiene que ser

   $$f \equiv \rho(c, \kappa \rho p S0S0 \kappa SpSSS0SSS0pSSS0SSS0pS0SSS0$$

   Más aún, como el primer funtor de la recursión es monádico, el segundo debe ser triádico.

2. Como el segundo funtor empieza por κ, debe ser una composición.

   (a) El primero de los funtores de la composición empieza por ρ, luego tiene que ser un funtor definido por recursión.

      i. El signo siguiente es p, luego el primer funtor de la recursión tiene que ser una proyección. Para que pueda serlo, tras la p deben venir dos numerales adecuados, y vemos que así es, que determinan la proyección p^1_1. Por lo tanto, la recursión debe completarse con un funtor triádico para formar un funtor diádico.

         $$f \equiv \rho(c, \kappa(\rho(p^1_1, \kappa SpSSS0SSS0pSSS0SSS0pS0SSS0$$

      ii. Tras p^1_1 viene una κ, luego el segundo funtor de la recursión tiene que ser una composición. Luego viene S, que es, pues, el primer funtor de la composición. Como es un funtor monádico, la composición deberá completarse con un único funtor.

         $$f \equiv \rho(c, \kappa(\rho(p^1_1, \kappa(S, pSSS0SSS0pSSS0SSS0pS0SSS0$$

      iii. A continuación de S viene una p, luego el segundo funtor de la composición tiene que ser una proyección. Vemos que se trata de p^3_3, y ahí tiene que acabar la composición y también la recursión que es, como debía ser, un funtor diádico:

         $$f \equiv \rho(c, \kappa(\rho(p^1_1, \kappa(S, p^3_3)), pSSS0SSS0pS0SSS0$$

   (b) Puesto que el primer funtor de la composición es diádico, ahora deben venir dos funtores más. El primero empieza por p, luego tiene que ser una proyección, y vemos que es p^3_3. A continuación tenemos otra p, que es el principio de p^3_1. Esto completa la composición:

      $$f \equiv \rho(c, \kappa(\rho(p^1_1, \kappa(S, p^3_3)), p^3_3, p^3_1)) \equiv \rho(c, \kappa(+, p^3_3, p^3_1)).$$

Si en algún momento no se hubiera cumplido algo de lo que hemos dicho que debía cumplirse, habríamos concluido que la cadena de signos dada no sería un funtor, pero, como todo ha cuadrado, concluimos que, en efecto, f es un funtor, que podemos construir mediante la sucesión:

$$\begin{array}{ll} f_1 & S \\ f_2 & p_3^3 \\ f_3 & \kappa(f_1, f_2) \\ f_3 & p_1^1 \\ f_5 & \rho(f_4, f_3) \\ f_6 & p_1^3 \\ f_7 & \kappa(f_5, f_2, f_6) \\ f_8 & c \\ f_9 & \rho(f_8, f_7) \end{array}$$

Comparando con los ejemplos de la sección precedente, concluimos que f no es sino la multiplicación usual de números naturales.

El lector debería convencerse de que lo que hemos hecho en este ejemplo se puede hacer en general: si nos dan una cadena de signos de $\mathcal{L}_{\text{arp}}$, siempre podemos determinar si es o no un funtor y, en caso afirmativo, obtener una sucesión de funtores que lo defina paso a paso. En realidad conviene probar algo ligeramente más general: dada una cadena de caracteres, siempre podemos determinar si empieza por un funtor, y en tal caso podemos determinar dónde termina éste.

Si el lector tiene conocimientos de programación, la mejor forma que tiene de convencerse de que esto es así es programar un algoritmo que haga esto precisamente, es decir, que, al darle una cadena de signos de $\mathcal{L}_{\text{arp}}$, determine si empieza o no por un funtor y, en caso afirmativo, que determine dónde termina y cuál es su estructura.

Al igual que sucedía con la definición de fórmula del lenguaje $\mathcal{L}_{\text{cp}}$, en la práctica nunca tendremos que preocuparnos por "leer" cadenas de signos presentadas tal cual, porque nunca las escribiremos así, al igual que no nos preocupamos por el hecho de que cualquier afirmación que hace normalmente un matemático sería prácticamente ilegible si la presentáramos sustituyendo en ella por sus definiciones todos los conceptos definidos.

Al igual que un matemático define unos conceptos a partir de otros previamente definidos, a nadie se le ocurriría definir la multiplicación de números naturales como

$$\cdot \equiv \rho c \kappa \rho p S 0 S 0 \kappa S p S S S 0 S S S 0 p S S S 0 S S S 0 p S 0 S S S 0.$$

En su lugar (y a falta de simplificaciones adicionales que veremos posteriormente), lo que hacemos es definir primero la suma y luego el producto, así:

$$+ \quad \equiv \quad \rho(p_1^1, \kappa(S, p_3^3)), \qquad \cdot \quad \equiv \quad \rho(c, \kappa(+, p_3^3, p_1^3)),$$

con lo que tenemos expresiones fácilmente analizables. Lo importante es que, a partir de ahora, cuando escribimos "$\cdot$", no estamos escribiendo una manchita de tinta que arbitrariamente hemos elegido para nombrar el producto de números naturales, sino que nos referimos a una cadena de signos de $\mathcal{L}_{\text{arp}}$ perfectamente determinada (aunque nadie esté interesado en verla "al natural") que forma parte de una "formulación general" que nos permite relacionar sistemáticamente dicha multiplicación con otras funciones que podemos definir a partir de ella. ∎

Ejemplo Vamos a analizar esta definición: $f \equiv \rho(\kappa(S,c),\kappa(\cdot,p^3_3,p^3_1))$. Se trata de un funtor que nombra la función $F = F(f)$ definida por recursión a partir de las funciones $c_1 = F(\kappa(S,c))$ y $H = F(\kappa(\cdot,p^3_3,p^3_1))$. La primera viene dada por

$$c_1(n) = F(S)(F(c_0)(n)) = F(S)(0) = 1,$$

mientras que la segunda es

$$H(m,n,r) = \cdot(p^3_3(m,n,r),p^3_1(m,n,r)) = r \cdot m.$$

Por lo tanto:

$$F(m,0) = 1, \qquad F(m,n+1) = H(m,n,F(m,n)) = F(m,n)\cdot m.$$

Sabemos que hay una única función que cumple estas dos propiedades y, a poca aritmética que sepa el lector, sabrá que las cumple la función exponencial $F(m,n) = m^n$, pues, en efecto:

$$m^0 = 1, \qquad m^{n+1} = m^n \cdot m.$$

En particular vemos que la exponenciación de números naturales es una función recursiva primitiva. ∎

Variables Llegados a este punto, ya tenemos una "nomenclatura" para todos los números naturales y para todas las funciones recursivas primitivas. Cada numeral nombra un número natural y cada funtor nombra una función recursiva primitiva, de modo que, viendo el numeral o el funtor, podemos saber qué número o qué función nombra. El paso siguiente es definir cadenas de signos de $\mathcal{L}_{\text{arp}}$ que se puedan interpretar como afirmaciones sobre los números naturales y las funciones recursivas primitivas. Para que podamos expresar propiedades como que

$$x \cdot (y+z) = x\cdot y + x \cdot z$$

necesitamos introducir "variables", es decir, cadenas x, y, z que hagan referencia, no a números naturales concretos, como sucede con los numerales, sino a números arbitrarios. Recordemos que el séptimo signo de $\mathcal{L}_{\text{arp}}$ es un signo x que todavía no hemos usado para nada:

Definición 1.4 Llamaremos *variables* a las cadenas de signos de $\mathcal{L}_{\text{arp}}$ formadas por el signo x seguido de un numeral, es decir:

$$x_0 \equiv x0, \quad x_1 \equiv xS0, \quad x_2 \equiv xSS0, \quad x_3 \equiv xSSS0, \quad \ldots$$

Diremos que x_i es la *variable de índice* i. Sin embargo, veremos que, cuando consideremos variables, nunca será relevante cuáles son sus índices, por lo que usaremos letras arbitrarias: $x, y, z, x_1, x_2, \ldots$ para referirnos a variables cualesquiera, sin que nunca vaya a importar cuáles son concretamente.

En principio, si estamos llamando x e y a dos variables, puede darse el caso de que $x \equiv y$, es decir, de que ambas sean la misma variable. No obstante, para evitar constantes precisiones, entenderemos que si estamos hablando de variables con nombres diferentes, se tratará de variables distintas, salvo que explícitamente indiquemos que podrían ser la misma. ∎

Términos Ahora ya podemos definir las cadenas de signos más generales que nombrarán números naturales en nuestro lenguaje. Damos directamente la definición formal en términos de sucesiones, analoga a la que hemos dado para los funtores:

Definición 1.5 Un *término* de $\mathcal{L}_{\text{arp}}$ es una cadena de signos t tal que exista una sucesión $t_1, \ldots, t_r$ de cadenas de signos de modo que $t_r \equiv t$ y, para cada i, se cumple uno de los casos siguientes:

1. $t_i \equiv 0$.
2. t_i es una variable.
3. $t_i \equiv f(t_{j_1}, \ldots, t_{j_n}) \equiv ft_{j_1}, \ldots, t_{j_n}$, donde f es un funtor n-ádico de $\mathcal{L}_{\text{arp}}$ y $j_1, \ldots, j_n < i$ son índices anteriores de la sucesión.

En la práctica, lo que realmente importa es que todo término t de $\mathcal{L}_{\text{arp}}$ se encuentra en uno de los casos siguientes:

1. $t \equiv 0$.
2. t es una variable.
3. $t \equiv f(t_1, \ldots, t_n) \equiv ft_1 \cdots t_n$, donde f es un funtor n-ádico y $t_1, \ldots, t_n$ son otros términos.

En otras palabras, los términos son las expresiones que se obtienen aplicando funtores al 0 y a las variables. En particular, todos los numerales —que resultan de aplicar el funtor S al 0— son términos.

Nota Como en el caso de los funtores, hay que señalar que los términos los definimos como meras yuxtaposiciones de funtores y términos, sin que sean necesarios paréntesis y comas (ajenos al lenguaje $\mathcal{L}_{\text{arp}}$) para determinar la estructura de un término.

Por ejemplo, si nos encontramos con:

$$t \equiv \rho pS0S0\kappa SpSSS0SSS0Sx0xS0,$$

ya hemos visto que podemos concluir que el principio de esta cadena de signos es el funtor $+ \equiv \rho(p_1^1, \kappa(S, p_3^3))$, de modo que tenemos

$$t \equiv +(Sx0xS0,$$

como la cadena empieza por un funtor diádico, para ser un término tiene que continuar con dos términos yuxtapuestos. El primero de dichos términos empieza por S, que es un funtor monádico, luego a continuación debe ir un término que complete el primer sumando, y luego el segundo sumando. Tras S encontramos x, luego el término que completa el primer sumando tiene que ser una variable, para lo cual es preciso que siga un numeral. Vemos que sigue el numeral 0, luego tenemos

$$t \equiv +(Sx_0, xS0.$$

El segundo sumando empieza por x, luego tiene que ser otra variable, para lo cual la x debe ir seguida de un numeral. En efecto, vemos que le sigue el numeral $S0$, por lo que llegamos a que la cadena de signos dada es el término

$$t \equiv +(Sx_0, x_1) \equiv (Sx) + y,$$

donde hemos introducido un último convenio de notación por el que, escribiremos $t_1 + t_2 \equiv +(t_1, t_2)$.

El lector puede comprobar que, en general, es posible programar un ordenador para que, dada una cadena de signos, determine si es o no un término y, en caso afirmativo, cuál es su estructura, a pesar de que en $\mathcal{L}_{\text{arp}}$ no haya paréntesis ni comas. ∎

Podemos introducir todos los convenios que consideremos oportunos para agilizar la lectura y la escritura de términos. Por ejemplo, igual que acabamos de convenir que $t_1 + t_2 \equiv +(t_1, t_2)$, podemos convenir que $t_1 \cdot t_2 \equiv \cdot(t_1, t_2)$ (e incluso podemos escribir $t_1 t_2$ cuando la omisión del punto no dé lugar a confusión).

Antes hemos visto que el funtor $f \equiv \rho(\kappa(S, c), \kappa(\cdot, p_3^3, p_1^3))$ denota la exponenciación de números naturales, y ahora podemos introducir el convenio de notación $t_1^{t_2} \equiv f(t_1, t_2)$, con lo que podemos considerar términos como

$$(x + y)^2, \qquad (x^2 + y^2) + 2(xy).$$

En principio, los paréntesis son necesarios, pues, por ejemplo,

$$(x^2 + y^2) + 2(xy) \equiv +(+(x^2, y^2), \cdot(2, xy)),$$

$$x^2 + (y^2 + (2x)y) \equiv +(x^2, +(y^2, \cdot(2x, y)))$$

son términos distintos.

Sin embargo, el lector interpretará de forma espontánea ambos términos como "el numero que resulta de multiplicar x por sí mismo, sumarle el producto de y por sí mismo y luego sumarle el doble del producto de x por y", por lo que, aunque no sean el mismo término, "deberían ser iguales". Ahora vamos a definir con precisión en qué consiste esto de "interpretar términos".

Definición 1.6 Una *valoración* es un criterio v que asigna a cada variable x de $\mathcal{L}_{\text{arp}}$ un número natural $v(x)$. Diremos que t *denota* un número natural N respecto de una valoración v si, dada una sucesión $t_1, \ldots, t_r$ de términos en las condiciones de la definición 1.5, existe una sucesión $N_1, \ldots, N_r$ de números naturales tal que $N_r = N$ y, para cada i, se cumpla:

1. Si $t_i \equiv 0$, entonces $N_i = 0$.
2. Si t_i es una variable, entonces $N_i = v(t_i)$.
3. Si $t_i \equiv f(t_{j_1}, \ldots, t_{j_n})$, entonces $N_i = F(f)(N_{j_1}, \ldots, N_{j_n})$.

Es fácil concluir entonces, como en el caso de la función denotada por un funtor, que para cada término t y cada valoración v de sus variables, existe un único número natural denotado por t respecto de v, y lo representaremos por $N(t)[v]$.

En la práctica basta tener en cuenta que la definición anterior implica los hechos siguientes:

1. $N(0)[v] = 0$.
2. Si x es una variable, $N(x)[v] = v(x)$.
3. $N(f(t_1, \ldots, t_n))[v] = F(f)(N(t_1)[v], \ldots, N(t_n)[v])$.

También es fácil concluir que $N(t)[v]$ depende a lo sumo de los valores que toma v sobre las variables que de hecho aparecen en t, por lo que, a la hora de calcular el número denotado por un término, basta asignar un valor a cada una de las variables que aparecen en él. En particular, si t no tiene variables, no es necesaria ninguna valoración y escribiremos simplemente $N(t)$.

Uniendo que $N(0) = 0$ con que $N(St)[v]$ es el siguiente del número natural $N(t)[v]$, una simple inducción prueba que el número natural denotado por un numeral es precisamente el número natural que nombra —o, mejor dicho, el número natural que pretendíamos que nombrara, ya que es ahora cuando estamos definiendo realmente el número natural denotado por un numeral—. Así pues:

$$N(0) = 0, \quad N(S0) = 1, \quad N(SS0) = 2, \quad N(SSS0) = 3, \quad \ldots$$

Por ejemplo, fijada una valoración v, tenemos que

$$N((x^2 + y^2) + 2(xy))[v] = N(x^2 + y^2)[v] + N(2(xy))[v],$$

donde hemos usado que $F(+)$ es la suma de números naturales. Por la misma razón, este número es

$$N(x^2)[v] + N(y^2)[v] + N(2(xy))[v].$$

Ahora usamos que $F(\cdot)$ es el producto de números naturales y que $F(()^{()})$ es la exponenciación, con lo que el número denotado por el término es

$$N(x)[v]^2 + N(y)^2[v] + N(2)[v] \cdot N(x)[v] \cdot N(y)[v].$$

Por último, usando que $N(2)[v] = 2$ y que $N(x)[v] = v(x)$, $N(y)[v] = v(y)$, llegamos a que

$$N((x^2 + y^2) + 2(xy))[v] = v(x)^2 + v(y)^2 + 2v(x)v(y).$$

En definitiva, aplicando la definición de denotación llegamos a que el término $(x^2 + y^2) + 2(xy)$ denota el número que a simple vista entendíamos que denotaba. En otras palabras, la definición de "denotación" que hemos dado no hace sino capturar lo que entendemos normalmente cuando leemos una expresión matemática. ∎

Fórmulas Hasta aquí hemos estudiado las cadenas de signos que nombran números naturales (los términos, al menos cuando se asignan valores a sus variables) y funciones (los funtores). Finalmente estamos en condiciones de definir cadenas de signos que expresen afirmaciones sobre los números naturales del estilo de

$$(x + y)^2 = (x^2 + y^2) + 2(xy).$$

Para ello metemos en juego al último signo de $\mathcal{L}_{\text{arp}}$, el que representamos por "=" y al que llamaremos *igualador:*

Definición 1.7 Una *fórmula* del lenguaje $\mathcal{L}_{\text{arp}}$ es una cadena de signos de la forma

$$t_1 = t_2 \quad \equiv \quad = t_1 t_2.$$

Los términos t_1 y t_2 se llaman *miembros* de la fórmula. Llamaremos *expresiones* a las cadenas de signos que son términos o fórmulas.

Así, "técnicamente", todas las fórmulas empiezan por el igualador, aunque en la práctica las nombraremos en la forma que se indica a la izquierda, con el igualador entre los dos miembros.

Notemos que los dos términos de los que consta una fórmula están unívocamente determinados, aunque estén yuxtapuestos, pues ya hemos señalado que si una cadena de signos $t_1 t_2$ es la yuxtaposición de dos términos, siempre es posible determinar unívocamente dónde termina el primero y dónde empieza el segundo.

Vemos que la cadena de signos

$$(x + y)^2 = (x^2 + y^2) + 2(xy)$$

es ciertamente una fórmula de $\mathcal{L}_{\text{arp}}$.

Si $\alpha \equiv (t_1 = t_2)$ es una fórmula y v es una valoración, diremos que α es *satisfecha* respecto a v, y lo representaremos por $\vDash \alpha[v]$, si se cumple

$$N(t_1)[v] = N(t_2)[v],$$

es decir, si los dos miembros denotan el mismo número natural cuando sus variables se interpretan según v. Notemos que esto sólo depende de cómo actúa v sobre las variables que están en α.

Por ejemplo, si $\alpha \equiv Sx = 5$, es claro que $\vDash (Sx = 5)[v]$ equivale a que $v(x) + 1 = 5$, luego también a que $v(x) = 4$.

Diremos que una fórmula α es *verdadera* si es satisfecha respecto de todas las valoraciones, y es *falsa* si no es satisfecha respecto de ninguna valoración. Usaremos la notación $\vDash \alpha$ para indicar que una fórmula es verdadera.

Nota En este punto es esencial tener presente que el hecho de que una fórmula α, digamos con variables $x_1, \ldots, x_n$, sea satisfecha o no por una valoración v depende únicamente de los valores $v(x_1), \ldots, v(x_n)$. Esto es importante porque podemos enumerar explícitamente todas las sucesiones de n números naturales $v_1, \ldots, v_n$.

Más precisamente, si α no tiene variables, entonces será verdadera o falsa según si es satisfecha o no por una valoración cualquiera (aunque está será del todo irrelevante), mientras que si tiene $n > 0$ variables, podemos programar a un ordenador para que nos proporcione una lista de todas las sucesiones $s_k = (v_1^k, \ldots, v_n^k)$ de n números naturales. Por ejemplo, para $n = 3$ una enumeración posible es

$$s_0 = (0,0,0), \quad s_1 = (1,0,0), \quad s_2 = (0,1,0), \quad s_3 = (0,0,1),$$
$$s_4 = (2,0,0), \quad s_5 = (1,1,0), \quad s_6 = (0,2,0), \quad s_7 = (1,0,1),$$
$$s_8 = (0,1,1), \quad s_9 = (0,0,2), \quad s_{10} = (3,0,0), \quad s_{11} = (2,1,0), \quad \ldots$$

El criterio es enumerar primero todas las sucesiones que cumplen la relación $v_1+v_2+v_3 = 0$ (y sólo hay una, s_0), luego todas las que cumplen $v_1+v_2+v_3 = 1$ (y hay tres), luego las que cumplen $v_1 + v_2 + v_3 = 2$ (hay seis), luego las que cumplen $v_1 + v_2 + v_3 = 3$, etc. Las sucesiones de cada grupo están ordenadas lexicográficamente (se compara primero el último número, en caso de empate se compara el penúltimo, etc.).

De este modo, que una fórmula α con variables x_1, x_2, x_3 sea verdadera significa que es satisfecha cuando $v(x_1) = 0, v(x_2) = 0, v(x_3) = 0$ (el caso correspondiente a s_0), y también cuando $v(x_1) = 1, v(x_2) = 0, v(x_3) = 0$ (el caso correspondiente a s_1), y también cuando $v(x_1) = 0, v(x_2) = 1, v(x_3) = 0$ (el caso correspondiente a s_2) y en general cuando las variables toman los valores determinados por la sucesión s_k, para todo k.

Más aún, podemos programar al ordenador para que, además de ir calculando las sucesiones s_k, vaya comprobando si α es satisfecha cuando sus variables se interpretan según s_k. En estos términos, la fórmula α será verdadera si el ordenador no encuentra nunca ninguna sucesión s_k con la que α no sea satisfecha.

Por supuesto, aunque pongamos al ordenador a calcular y veamos que va obteniendo que α es satisfecha con todas las sucesiones que va calculando, en principio no tenemos forma de saber si al cabo de un tiempo encontrará una sucesión para la cual deje de ser así, o si jamás podría darse el caso, es decir, no tenemos garantías de que podamos averiguar si una fórmula es o no verdadera, pero lo importante es que sabemos lo que significa que α es verdadera (tanto si lo podemos comprobar o no). ∎

Conviene destacar también el matiz que hemos introducido al distinguir entre verdad y satisfacción: un matemático consideraría natural decir que la fórmula $Sx = 5$ es verdadera cuando $x = 4$ y falsa en otro caso, pero nosotros diremos que la fórmula es satisfecha cuando asignamos a x el valor $v(x) = 4$ y no es satisfecha en caso contrario, pero la fórmula no es ni verdadera ni falsa.

En la práctica, una fórmula es verdadera si lo que se entiende al leerla es verdad (para cualquier interpretación de las variables). Por ejemplo, la fórmula

$$(x + y)^2 = (x^2 + y^2) + 2(xy)$$

es verdadera, pero no hace falta recurrir a las definiciones de denotación y verdad para comprobarlo, pues a simple vista se ve que lo que afirma es que, para toda valoración v, se cumple

$$(v(x) + v(y))^2 = v(x)^2 + v(y)^2 + 2v(x)v(y),$$

y eso es cierto. ∎

Definiciones de funtores Hasta ahora hemos asociado a cada funtor un significado (la función que denota), pero eso es algo ajeno al propio lenguaje $\mathcal{L}_{\text{arp}}$, en el sentido de que las afirmaciones del tipo "el funtor tal denota tal función" no son fórmulas de $\mathcal{L}_{\text{arp}}$. Ahora estamos en condiciones de expresar la definición de cada funtor de $\mathcal{L}_{\text{arp}}$ distinto de S mediante fórmulas adecuadas de $\mathcal{L}_{\text{arp}}$.

Definición 1.8 A cada funtor de $\mathcal{L}_{\text{arp}}$ distinto de S le vamos a asignar una o dos fórmulas de $\mathcal{L}_{\text{arp}}$ a las que llamaremos su *definición* (o las *fórmulas que definen el funtor*). Concretamente:

- La definición del funtor c es la fórmula[2] $c(x) = 0$.
- La definición del funtor p^n_k es la fórmula[3] $p^n_k(x_1, \ldots, x_n) = x_k$.
- La definición de un funtor $f \equiv \kappa(h, g_1, \ldots, g_m)$, donde h es m-ádico y los g_i son n-ádicos, es la fórmula

$$f(x_1, \ldots, x_n) = h(g_1(x_1, \ldots, x_n), \ldots, g_m(x_1, \ldots, x_n)).$$

[2]Podríamos especificar qué variable en particular es x, pero veremos que será irrelevante.
[3]Con el convenio de que nombres distintos nombran a variables distintas.

- La definición de un funtor $f \equiv \rho(g,h)$ son las fórmulas

$$f(x_1,\ldots,x_n,0) = g(x_1,\ldots,x_n),$$
$$f(x_1,\ldots,x_n,Sx_{n+1}) = h(x_1,\ldots,x_n,x_{n+1},f(x_1,\ldots,x_n,x_{n+1})).$$

Es inmediato que las fórmulas que definen los funtores son verdaderas por la propia definición de la función denotada por un funtor y de la verdad de una fórmula.

Observemos que no es posible dar una definición del funtor S, pues el hecho de que Sx sea el número natural siguiente a x no puede expresarse en términos más elementales.

En la práctica, a partir de la definición de un funtor es fácil determinar de qué funtor estamos hablando, por lo que en realidad nunca definiremos funtores usando κ o ρ, sino a través de las fórmulas que los definen en $\mathcal{L}_{\mathrm{arp}}$. Por ejemplo, en lugar de presentar la suma como $+ \equiv \quad \rho(p^1_1, \kappa(S, p^3_3))$, diremos simplemente que $+$ es el funtor definido por las ecuaciones

$$\begin{aligned} m+0 &= m \\ m+Sn &= S(m+n). \end{aligned}$$

A partir de ellas es claro que estamos definiendo un funtor diádico de la forma $\rho(g,h)$, donde necesariamente $g(m) = m$, luego tiene que ser $g \equiv p^1_1$, y $h(m,n,r) = Sr$, luego tiene que ser $h \equiv \kappa(S, p^3_3)$.

Similarmente, el producto puede definirse como el único funtor que cumple

$$\begin{aligned} m\cdot 0 &= 0 \\ m\cdot Sn &= m\cdot n + m. \end{aligned}$$

Nuevamente, esto implica que el funtor definido tiene que ser un funtor diádico de la forma $\rho(g,h)$, donde $g(m) = 0$, luego tiene que ser $g \equiv c$, y $h(m,n,r) = r + m$, luego $h \equiv \kappa(+, p^3_3, p^3_1)$, lo que nos lleva a

$$\cdot \equiv \rho(c, \kappa(+, p^3_3, p^3_1)).$$

Sustitución Necesitamos describir una manipulación elemental que podemos llevar a cabo con las expresiones de $\mathcal{L}_{\mathrm{arp}}$. Si tenemos, por ejemplo, una fórmula como

$$(x+y)(x+y) = (x\cdot x + 2\cdot(x\cdot y)) + y\cdot y,$$

a partir de ella podemos construir la fórmula que resulta de sustituir una de sus variables, por ejemplo la y, por un término cualquiera, por ejemplo, $t \equiv 3\cdot z$. El resultado es

$$\mathtt{S}^{3\cdot z}_y((x+y)(x+y) = (x\cdot x + 2\cdot(x\cdot y)) + y\cdot y) \equiv$$
$$(x+(3\cdot z))(x+(3\cdot z)) = (x\cdot x + 2\cdot(x\cdot(3\cdot z))) + (3\cdot z)\cdot(3\cdot z).$$

En general:

Definición 1.9 Si θ es una expresión de $\mathcal{L}_{\text{arp}}$, x es una variable y t es un término, llamaremos *sustitución de x por t en θ,* y la representaremos por $\mathtt{S}^t_x\theta$, a la expresión que resulta de sustituir cada aparición de la variable x en θ por el término t.

Observemos que $\mathtt{S}^t_x\theta$ es ciertamente una expresión (y, más precisamente, es un término o una fórmula según lo sea θ). Esto se pone de manifiesto si expresamos la sustitución en términos de la estructura de θ:

1. $S^t_x 0 \equiv 0$,
2. $S^t_x y \equiv \begin{cases} t & \text{si } x \equiv y, \\ y & \text{en caso contrario,} \end{cases}$
3. $\mathtt{S}^t_x f(t_1,\dots,t_n) \equiv f(\mathtt{S}^t_x t_1,\dots,\mathtt{S}^t_x t_n)$,
4. $\mathtt{S}^t_x(t_1 = t_2) \equiv S^t_x t_1 = S^t_x t_2$.

Si v es una valoración, x es una variable y n es un número natural, llamaremos v^n_x a la valoración que coincide con v salvo por que a x le asigna el valor n. Entonces:

Teorema 1.10 *El término $S^t_x t'$ denota, respecto de una valoración v, el mismo número natural que t' denota respecto de la valoración $v^{N(t)[v]}_x$. Equivalentemente:*

$$N(\mathtt{S}^t_x t')[v] = N(t')[v^{N(t)[v]}_x].$$

Si α es una fórmula, se cumple $\vDash \mathtt{S}^t_x\alpha[v]$ si y sólo si $\vDash \alpha[v^{N(t)[v]}_x]$.

Esto significa que el significado de $\mathtt{S}^t_x t'$ o de $\mathtt{S}^t_x\alpha$ es el significado de t' o α cuando la variable x se interpreta con el significado de t.

Demostración: Si $t' \equiv 0$ se cumple la igualdad:

$$N(\mathtt{S}^t_x 0)[v] = N(0)[v] = 0, \qquad N(0)[v^{N(t)[v]}_x] = 0.$$

Si t' es una variable distinta de x también:

$$N(\mathtt{S}^t_x t')[v] = N(t')[v], \qquad N(t')[v^{N(t)[v]}_x] = N(t')[v].$$

Si $t' \equiv x$ entonces

$$N(\mathtt{S}^t_x x)[v] = N(t)[v], \qquad N(x)[v^{N(t)[v]}_x] = N(t)[v].$$

Si el resultado fuera falso, podríamos tomar un término t' que lo incumpla de longitud mínima. Según acabamos de ver, no puede ser que t' sea 0 o una variable, luego $t' \equiv f(t_1,\dots,t_n)$, para ciertos términos t_i que cumplen la conclusión por la minimalidad de t'. Entonces

$$N(\mathtt{S}^t_x f(t_1,\dots,t_n))[v] = N(f(\mathtt{S}^t_x t_1,\dots,\mathtt{S}^t_x t_n))[v] =$$

$$N(f)(N(\mathtt{S}^t_x t_1)[v],\dots N(\mathtt{S}^t_x t_n)[v]) = N(f)(N(t_1)[v^{N(t)[v]}_x],\dots,N(t_n)[v^{N(t)[v]}_x]) =$$

$$N(f(t_1,\dots,t_n))[v^{N(t)[v]}_x],$$

con lo que t' también cumple el resultado y tenemos una contradicción.

Si $\alpha \equiv t' = t''$, entonces $\vDash \mathsf{S}^t_x\alpha[v]$ equivale a

$$N(\mathsf{S}^t_x t')[v] = N(\mathsf{S}^t_x t'')[v],$$

que a su vez equivale a $N(t')[v_x^{N(t)[v]}] = N(t'')[v_x^{N(t)[v]}]$, y esto a su vez equivale a $\vDash \alpha[v_x^{N(t)[v]}]$. ∎

Así pues, $\mathsf{S}^t_x\alpha$ dice de t lo que α dice de x.

Una notación muy práctica consiste en escribir $\theta(x_1,\ldots,x_n)$ para referirnos a una expresión θ, donde $x_1,\ldots,x_n$ son variables cualesquiera, que pueden estar o no en θ, y luego entender que $\theta(t_1,\ldots,t_n)$ es la expresión que resulta de sustituir cada variable x_i por el término t_i. Pero aquí hay un problema técnico que obliga a adoptar la definición siguiente:

$$\theta(t_1,\ldots,t_n) \equiv \mathsf{S}^{t_1}_{y_1}\cdots\mathsf{S}^{t_n}_{y_n}\mathsf{S}^{y_1}_{x_1}\cdots\mathsf{S}^{y_n}_{x_n}\theta,$$

donde $y_1,\ldots,y_n$ son variables cualesquiera que no estén en $t_1,\ldots,t_n$, ni en θ, ni en $x_1,\ldots,x_n$.

Para entender por qué es necesario introducir las variables $y_1,\ldots,y_n$ consideremos $\theta(x,y) \equiv x = y$. Por ejemplo, queremos que

$$\theta(S0, x+y) \equiv S0 = x+y,$$

pero si hubiéramos definido

$$\theta(S0, x+y) \equiv \mathsf{S}^{S0}_x\mathsf{S}^{x+y}_y\theta,$$

tendríamos que

$$\theta(S0, x+y) \equiv \mathsf{S}^{S0}_x(x = x+y) \equiv S0 = S0+y,$$

que no es lo que queremos.

En cambio, con la definición que hemos dado:

$$\theta(S0, x+y) \equiv \mathsf{S}^{S0}_{y_1}\mathsf{S}^{x+y}_{y_2}\mathsf{S}^{y_1}_x\mathsf{S}^{y_2}_y(x=y) \equiv \mathsf{S}^{S0}_{y_1}\mathsf{S}^{x+y}_{y_2}\mathsf{S}^{y_1}_x(x=y_2)$$

$$\equiv \mathsf{S}^{S0}_{y_1}\mathsf{S}^{x+y}_{y_2}(y_1=y_2) \equiv \mathsf{S}^{S0}_{y_1}(y_1 = x+y) \equiv S0 = x+y,$$

como debe ser. Vemos así que al introducir las variables auxiliares $y_1,\ldots,y_n$ (que luego se eliminan) evitamos que algún t_i pase a ocupar el lugar que x_i ocupa en algún término t_j sustituido previamente.

Consideraciones finales Con esto tenemos ya convenientemente descrito el lenguaje $\mathcal{L}_{\mathrm{arp}}$. Ahora podemos ver

$$x^{y+z} = x^y \cdot x^z$$

y pensar que es una afirmación válida para tres números naturales arbitrarios, pero al mismo tiempo afirmar que no es más que una cadena de signos de $\mathcal{L}_{\mathrm{arp}}$ que podemos manipular formalmente sin tener nunca en cuenta que significa algo.

Mientras al razonar intuitivamente tenemos que cerciorarnos de que nunca decimos algo que no tenga un significado preciso, esta precaución sobra cuando expresamos un hecho a través de una fórmula de $\mathcal{L}_{\text{arp}}$, pues cada fórmula de $\mathcal{L}_{\text{arp}}$ tiene siempre un único significado posible.

El lector también debe reflexionar sobre cómo hemos definido los conceptos de "numeral", "funtor", "termino", "fórmula", etc. sin hacer referencia para nada a su significado a pesar de que hemos asignado un significado a cada numeral, a cada funtor, a cada término y a cada fórmula. Esto puede resultar anecdótico en este momento, pero el hecho de que podamos trabajar con lenguajes formales prescindiendo completamente del posible significado de sus términos y sus fórmulas se vuelve esencial cuando se trata de diseñar lenguajes formales para hablar sobre conceptos abstractos a los que no podemos asignarles un significado intuitivo preciso. ∎

1.2 Demostraciones en ARP

Hemos visto que cada funtor de ARP distinto de S admite una definición en ARP, es decir, que existen unas fórmulas de ARP (una sola fórmula, salvo en el caso de los funtores definidos por recursión, que requieren dos) que expresan en $\mathcal{L}_{\text{arp}}$ el significado del funtor que definen. Sin embargo, hay tres signos de ARP a los que hemos asignado un significado muy concreto, pero que no hemos expresado —ni se puede expresar— en términos de fórmulas de $\mathcal{L}_{\text{arp}}$. Se trata de los signos $0, S, =$. No podemos expresar mediante una fórmula de $\mathcal{L}_{\text{arp}}$ qué es el 0, ni qué es el siguiente de un número natural, ni qué es "ser igual", ni tampoco que, en una fórmula, x significa "para todo número natural x" . En vez de tratar de definirlos, vamos a dar unas reglas de razonamiento que nos permitan afirmar cosas que involucren estos signos sin necesidad de una definición.

Diremos que una fórmula α de ARP es *consecuencia lógica* de una o varias fórmulas $\alpha_1, \ldots, \alpha_n$ de ARP si alguna de las fórmulas α_i es falsa o bien α es verdadera. Lo expresaremos así:

$$\frac{\alpha_1, \ldots, \alpha_n}{\alpha}$$

Alternativamente, al afirmar que α es consecuencia de $\alpha_1, \ldots, \alpha_n$ estamos afirmando que, si $\alpha_1, \ldots, \alpha_n$ son verdaderas, α también tiene que serlo.

He aquí la regla de razonamiento que expresa formalmente nuestro convenio de considerar que las variables hacen referencia a números naturales arbitrarios. La llamaremos *primera regla de sustitución* y podemos expresarla así:

$$(S_1)\ \frac{\alpha}{\mathtt{S}^t_x\alpha}$$

o, más explícitamente, así:

$$(S_1)\ \frac{s_1 = s_2}{\mathtt{S}^t_x s_1 = \mathtt{S}^t_x s_2}$$

Esto significa que si una formula α (que contiene la variable x o, de lo contrario la conclusión es la propia α) es verdadera, también lo es la fórmula que resulta de sustituir el número natural genérico x por cualquier término t (que si α vale para todo x en particular vale para $x = t$). Un caso particular es, por ejemplo,

$$\frac{x + Sy = S(x + y)}{x + 1 = S(x + 0)}$$

donde hemos usado que $S0 \equiv 1$.

Vamos a razonar que, en efecto, si se cumple $\vDash s_1 = s_2$, necesariamente se cumple también $\vDash \mathtt{S}^t_x s_1 = \mathtt{S}^t_x s_2$.

Esto significa que, para cualquier valoración v, tiene que cumplirse

$$\vDash (\mathtt{S}^t_x s_1 = \mathtt{S}^t_x s_2)[v],$$

y, por el teorema 1.10, esto equivale a que

$$\vDash (s_1 = s_2)[v^{N(t)[v]}_x],$$

y esto se sigue de que $\vDash s_1 = s_2$, pues entonces la fórmula $s_1 = s_2$ es satisfecha por cualquier valoración, en particular por la valoración $v^{N(t)[v]}_x$. ∎

Ahora presentamos dos reglas de inferencia que expresan el significado del igualador. Son la *segunda regla de sustitución* y la *regla de transitividad:*

$$(S_2)\ \frac{t_1 = t_2}{\mathtt{S}^{t_1}_x s = \mathtt{S}^{t_2}_x s} \qquad (T)\ \frac{t_1 = t_2 \qquad t_1 = t_3}{t_2 = t_3}$$

La segunda regla de sustitución expresa que si dos términos son iguales, cualquier cálculo s que hagamos con $x = t_1$ dará el mismo resultado que si lo hacemos con $x = t_2$. Por ejemplo:

$$\frac{x + 0 = x}{S(x + 0) = Sx}$$

Para probar la validez de esta regla hemos de probar que, si $\vDash t_1 = t_2$, necesariamente $\vDash \mathtt{S}^{t_1}_x s = \mathtt{S}^{t_2}_x s$. Para ello tomamos una valoración v y tenemos que probar que

$$N(\mathtt{S}^{t_1}_x s)[v] = N(\mathtt{S}^{t_2}_x s)[v],$$

pero al estudiar la sustitución hemos probado que esto equivale a

$$N(s)[v^{N(t_1)[v]}_x] = N(s)[v^{N(t_2)[v]}_x],$$

pero la hipótesis $\vDash t_1 = t_2$ implica que $N(t_1)[v] = N(t_2)[v]$, luego en efecto tenemos la igualdad requerida. ∎

El significado de la regla de transitividad es obvio. Por ejemplo:

$$\frac{S(x+0)=x+1 \qquad S(x+0)=Sx}{x+1=Sx}$$

La prueba de su validez es trivial: si $\vDash t_1 = t_2$ y $\vDash t_1 = t_3$, entonces, para toda valoración v, se cumple que $N(t_1)[v] = N(t_2)[v]$, y $N(t_1)[v] = N(t_3)[v]$, luego también $N(t_2)[v] = N(t_3)[v]$, y esto significa que $\vDash t_2 = t_3$. ∎

Veremos que estas dos reglas de inferencia son suficientes para que podamos "olvidar" que el igualador $=$ se interpreta como "es igual a" y, pese a ello, podamos razonar pasando de unas fórmulas a otras mediante reglas de inferencia, pero antes vamos a analizar una cuarta regla que nos permitirá "olvidar" también el significado de 0 y del funtor S. Se trata de una regla que expresa el principio de inducción:

$$(I_0)\ \frac{s_1(0)=s_2(0) \qquad s_1(Sx)=h(x,s_1(x)) \qquad s_2(Sx)=h(x,s_2(x))}{s_1(x)=s_2(x)}$$

Más precisamente, la regla afirma que si dos términos coinciden en $x = 0$ y los valores de $s_1(Sx)$ y $s_2(Sx)$ se pueden calcular a partir de x y $s_1(x)$ o x y $s_2(x)$, respectivamente mediante un mismo cálculo representado por el término h, entonces $s_1(x) = s_2(x)$ para cualquier valor de x. Un caso particular sería:

$$\frac{(x+y)+0=x+(y+0) \quad (x+y)+Sz=S((x+y)+z) \quad x+(y+Sz)=S(x+(y+z))}{(x+y)+z=x+(y+z)}$$

En efecto, supongamos que queremos probar que la fórmula

$$(x+y)+z=x+(y+z)$$

es verdadera. Admitiendo la validez de la regla de inducción, basta razonar que es verdadera cuando $z = 0$ (la primera premisa), y que $(x+y)+Sz$ y $x+(y+Sz)$ pueden calcularse, respectivamente, a partir de $(x+y)+z$ y $x+(y+z)$ mediante el mismo cálculo (en este caso, pasando al siguiente, como afirman las otras dos premisas).

Vamos a justificar que esta regla es válida en general. Para ello suponemos

$$\vDash \mathtt{S}^0_x s_1 = \mathtt{S}^0_x s_2, \quad \vDash \mathtt{S}^{Sx}_x s_1 = \mathtt{S}^{s_1}_y h, \quad \vDash \mathtt{S}^{Sx}_x s_2 = \mathtt{S}^{s_2}_y h$$

y tenemos que razonar que $\vDash s_1 = s_2$.

Fijamos una valoración v y tenemos que probar que

$$N(s_1)[v] = N(s_2)[v].$$

Vamos a ver que, para todo n, se cumple $N(s_1)[v^n_x] = N(s_2)[v^n_x]$, con lo que para $n = v(x)$ tendremos la conclusión. La primera premisa nos da que

$$N(\mathtt{S}^0_x s_1)[v] = N(\mathtt{S}^0_x s_2)[v],$$

lo que, por 1.10, equivale a que

$$N(s_1)[v_x^0] = N(s_2)[v_x^0],$$

Ahora supongamos que

$$N(s_1)[v_x^n] = N(s_2)[v_x^n].$$

La segunda premisa nos da que

$$N(\mathtt{S}_x^{Sx} s_1)[v_x^n] = N(\mathtt{S}_y^{s_1} h)[v_x^n],$$

lo que equivale a

$$N(s_1)[v_x^{N(Sx)[v_x^n]}] = N(h)[v_{x,y}^{n,N(s_1)[v_x^n]}],$$

pero $N(Sx)[v_x^n] = N(x)[v_x^n] + 1 = v_x^n(x) + 1 = n + 1$, luego tenemos

$$N(s_1)[v_x^{n+1}] = N(h)[v_{x,y}^{n,N(s_1)[v_x^n]}].$$

Igualmente, de la tercera premisa llegamos a

$$N(s_2)[v_x^{n+1}] = N(h)[v_{x,y}^{n,N(s_2)[v_x^n]}].$$

Teniendo en cuenta la hipótesis de inducción, concluimos que

$$N(s_1)[v] = N(s_2)[v],$$

como había que probar. ∎

Ahora estamos en condiciones de dar una definición de "razonamiento en ARP" puramente formal, es decir, que no dependa para nada del significado de las fórmulas que intervengan en el razonamiento. Para ello nos basaremos en la definición siguiente:

Reglas de inferencia Llamaremos *reglas de inferencia (primitivas)* de la Aritmética Recursiva Primitiva a los cuatro criterios siguientes, que definen cuándo una fórmula de ARP es *consecuencia inmediata* de otra u otras fórmulas dadas:

$$(S_1)\ \frac{s_1(x) = s_2(x)}{s_1(t) = s_2(t)} \qquad (S_2)\ \frac{t_1 = t_2}{s(t_1) = s(t_2)} \qquad (T)\ \frac{t_1 = t_2 \qquad t_1 = t_3}{t_2 = t_3}$$

$$(I_0)\ \frac{s_1(0) = s_2(0) \qquad s_1(Sx) = h(x, s_1(x)) \qquad s_2(Sx) = h(x, s_2(x))}{s_1(x) = s_2(x)}$$

Aquí $s(x)$, $s_1(x)$, $s_2(x)$, t, t_1, t_2, t_3 y $h(x, y)$ son términos arbitrarios.

Acabamos de probar que las reglas de inferencia de ARP son reglas de inferencia lógicas, en el sentido de que sus conclusiones son necesariamente verdaderas cuando lo son sus premisas, pero eso es algo que podemos "olvidar". Lo único que acabamos de hacer es tomar como definición que una fórmula de tipo $s_1(t) = s_2(t)$ es —por definición— consecuencia inmediata en ARP de la premisa $s_1(x) = s_2(x)$, y lo mismo en los otros tres casos. Lo que aquí es crucial es que, para determinar si una fórmula dada es o no consecuencia inmediata en ARP de otras fórmulas dadas no es necesario investigar si éstas son verdaderas o falsas (o nada), sino que es suficiente comprobar si su estructura sintáctica encaja o no con una de las cuatro reglas anteriores.

Definición 1.11 Llamaremos *axiomas* de la Aritmética Recursiva Primitiva a las fórmulas descritas en la definición 1.8, que definen a los funtores de ARP distintos de S.

Si Γ es un conjunto de fórmulas de ARP, una *deducción* en ARP con Γ como conjunto de *premisas* es una sucesión $\alpha_1, \ldots, \alpha_n$ de fórmulas de ARP tales que cada una de ellas sea, o bien un axioma, o bien una premisa, o bien sea consecuencia inmediata de fórmulas anteriores de la sucesión.

Una *demostración* en ARP es una deducción sin premisas.

Diremos que una fórmula α es una *consecuencia* en ARP de unas premisas $\alpha_1, \ldots, \alpha_n$ si existe una deducción con dichas premisas cuya última línea es α. Lo expresaremos así:

$$\alpha_1, \ldots, \alpha_n \underset{\text{ARP}}{\vdash} \alpha$$

o también así:

$$\frac{\alpha_1, \ldots, \alpha_n}{\alpha}$$

A cualquier resultado de este tipo lo llamaremos *regla de inferencia derivada* en ARP con las *premisas* y la *conclusión* indicadas.

Una fórmula α es un *teorema* de ARP si existe una demostración en ARP que tenga a α como última línea. Lo expresaremos así: $\underset{\text{ARP}}{\vdash} \alpha$.

Es claro que en una deducción en ARP podemos incluir cualquier fórmula α que sea consecuencia en ARP de fórmulas anteriores de la deducción, aunque no sea una consecuencia inmediata, pues siempre podríamos extender la deducción intercalando las fórmulas que justifican que α se deduce de tales fórmulas anteriores. En particular, en toda deducción (o demostración) en ARP se pueden incluir teoremas ya demostrados sin necesidad de repetir cada vez su demostración.

Enseguida veremos ejemplos que ilustren todos estos conceptos, pero conviene señalar que con esto hemos completado la definición de la Aritmética Recursiva Primitiva. Si el lector se pregunta qué es, concretamente, ARP, la respuesta es que no es nada en particular. Simplemente, podríamos haber definido otro lenguaje formal con otras características, o haber tomado otros axiomas u

otras reglas de inferencia primitivas, y en tal caso tendríamos otros conceptos distintos de lo que es un teorema, etc. Cuando realizamos deducciones en el lenguaje de ARP, a partir de los axiomas de ARP y usando las reglas de inferencia de ARP estamos razonando (formalmente) en ARP, por oposición a lo que sucedería si usáramos otro lenguaje formal, otros axiomas u otras reglas de inferencia, y eso es ARP.

El teorema de corrección De acuerdo con las definiciones que hemos dado, el hecho de que una sucesión de cadenas de signos de ARP sea o no una deducción (o una demostración) en ARP depende únicamente de la forma de esas cadenas, es decir, de la forma en que se disponen sus signos, cosa que puede comprobar un ordenador sin más que ir analizando dichos signos sin necesidad de investigar en ningún momento si las cadenas en cuestión son o no fórmulas verdaderas. Dicho con otras palabras, uno puede comprobar que un razonamiento en ARP es correcto sin necesidad de entender ninguna de las fórmulas que aparecen en él. Sólo tiene que comprobar si están en la lista de axiomas o premisas o si cuadran con alguna de las cuatro reglas de inferencia primitivas.

Con esto hemos desvinculado completamente la noción de "razonamiento" de la noción de "verdad". No obstante, por otra parte hemos razonado que todos los axiomas de ARP son fórmulas verdaderas y que las cuatro reglas de inferencia primitivas son lógicamente válidas, es decir, que sus conclusiones son verdaderas siempre que sus premisas lo son. Esto vuelve inmediato el teorema siguiente:

Teorema 1.12 (Teorema de corrección) *La conclusión de una deducción en* ARP *es necesariamente una fórmula verdadera si sus premisas lo son. En particular, todos los teoremas de* ARP *son fórmulas verdaderas.*

DEMOSTRACIÓN: Si $\alpha_1, \ldots, \alpha_n$ es una deducción en ARP, podemos razonar que cada α_i es una fórmula verdadera, luego en particular lo será su conclusión α_n. En efecto, en caso contrario podríamos considerar el menor índice i tal que α_i no fuera verdadera, pero entonces α_i no puede ser una premisa (porque estamos suponiendo que las premisas son verdaderas), ni tampoco un axioma (porque hemos razonado que todos los axiomas son verdaderos), luego α_i tiene que ser consecuencia inmediata de fórmulas anteriores de la deducción. Pero, por la minimalidad de i, dichas fórmulas tienen que ser verdaderas, y sabemos que las conclusiones de las reglas de inferencia son verdaderas cuando sus premisas lo son, por lo que α_i también tiene que ser verdadera, y así tenemos una contradicción. ∎

Así pues, aunque al razonar en ARP no necesitamos preocuparnos de si las fórmulas que consideramos son verdaderas o falsas (o nada), lo cierto es que tenemos la garantía de que todos los teoremas que demostremos serán fórmulas verdaderas sobre la aritmética de los números naturales.

Teoremas y metateoremas Conviene observar en este punto que estamos usando la palabra "teorema" en dos sentidos distintos. Por una parte, hemos

definido un teorema de ARP como una fórmula de $\mathcal{L}_{\text{arp}}$ demostrable sin premisas, mientras que el teorema de corrección, o el más modesto teorema 1.10, no son teoremas en este sentido, no son fórmulas de $\mathcal{L}_{\text{arp}}$ demostrables a partir de los axiomas de ARP mediante las reglas de inferencia de ARP.

Por el contrario, el teorema de corrección es lo que se suele llamar también un *metateorema,* es decir un teorema sobre una teoría formal, y en este caso es una afirmación enunciada en castellano (ampliado con una jerga técnica) y demostrada informalmente, es decir, que, para convencerse de que la demostración es correcta, el lector no puede buscar axiomas ni aplicaciones de reglas de inferencia, sino que meramente tiene que convencerse de que cada cosa que se dice tiene un significado preciso y es verdad.

Más precisamente, la demostración del teorema de corrección es un argumento genérico y el lector tiene que convencerse de que será aplicable siempre que tengamos ante nosotros una demostración concreta en ARP. Ésta constará de una sucesión de fórmulas y la prueba del teorema nos dice qué tenemos que hacer para convencernos de que cada una de ellas será verdadera, y la clave es que lo que nos dice que tenemos que hacer es algo que siempre se puede hacer, y de ahí la universalidad de su conclusión.

Por supuesto, 1.10 y el teorema de corrección no son, ni mucho menos, los únicos metateoremas que hemos demostrado hasta ahora, sino que todas las afirmaciones que hemos hecho sobre ARP lo son, aunque no las hayamos destacado tanto, como que la sustitución de una variable por un término en una fórmula es una fórmula, etc.

El teorema de corrección ilustra también que la lógica matemática no se limita a definir teorías axiomáticas y a trabajar en ellas, sino que permite estudiar si las teorías consideradas cumplen lo que se espera de ellas. En principio, podríamos haber definido otra teoría axiomática con otros axiomas y otras reglas de inferencia diferentes de los que hemos elegido para ARP y a ojos de un formalista no sería más que una teoría alternativa, pero si demostráramos que en dicha variante se pueden demostrar fórmulas falsas, entonces quedaría claro que dicha variante no sería lo que andamos buscando. El teorema de corrección muestra que la definición de ARP va en la dirección correcta.

En este punto el formalista reconvertido que ha asumido a regañadientes que necesita razonamientos informales para fundamentar la matemática formal tiene que debatirse entre el placer de haber encontrado una teoría axiomática formal en la que puede razonar exigiendo los estándares de rigor a los que está acostumbrado (y con la tranquilidad de que no necesita preocuparse de nada más que de respetarlos) y la zozobra de tener que abandonar este pequeño paraíso para razonar informalmente sobre ARP para demostrar los metateoremas como el teorema de corrección y otros que veremos a continuación, que nos proporcionarán técnicas de razonamiento formal en ARP que no son evidentes a partir de la mera definición de demostración formal. ∎

Sucede que la potencia de ARP, es decir, su capacidad para enunciar y demostrar resultados aritméticos, es mucho mayor de la que podría parecer a partir de la definición que hemos dado, pero para mostrar las posibilidades

reales de ARP necesitamos demostrar, más o menos "penosamente", algunos resultados básicos, de modo que, en cuanto dispongamos de un cierto número de estos resultados, el trabajo en ARP será mucho más sencillo y natural.

El igualador Como primeros ejemplos de demostraciones y deducciones en ARP vamos a probar algunos resultados sobre el igualador. Nuestro primer resultado es el siguiente: si t es cualquier término, entonces

$$\underset{\text{ARP}}{\vdash} t = t$$

Para probarlo, elegimos una variable cualquiera x y razonamos así:

(1)	$p_1^1(x) = x$	Definición de p_1^1
(2)	$p_1^1(t) = t$	S_1, 1
(3)	$t = t$	T, 2, 2

Notemos que hemos aplicado la regla de inferencia de transitividad T tomando como sus dos premisas la misma línea 2 de la demostración (alternativamente, podríamos haber deducido (2) dos veces a partir de (1)).

Ahora mostramos un ejemplo de regla derivada de inferencia, a la que llamaremos regla de *simetría:*

$$(S)\,\frac{t_1 = t_2}{t_2 = t_1}$$

Esto significa que tomando $t_1 = t_2$ como premisa podemos deducir $t_2 = t_1$. Una deducción es la siguiente:

(1)	$t_1 = t_2$	Premisa
(2)	$t_1 = t_1$	Teorema
(3)	$t_2 = t_1$	T, 1, 2

De aquí deducimos a su vez una variante de la regla de transitividad:

$$(T)\,\frac{t_1 = t_2 \qquad t_2 = t_3}{t_1 = t_3}$$

La deducción es:

(1)	$t_1 = t_2$	Premisa
(2)	$t_2 = t_1$	S, 1
(3)	$t_2 = t_3$	Premisa
(4)	$t_1 = t_3$	T, 2, 3

En la práctica llamaremos (T) a cualquiera de las dos reglas de transitividad, aunque una sea una regla primitiva y la otra una regla derivada.

Ahora podemos probar una variante de S_2 que muestra más claramente que dos términos iguales cumplen lo mismo:

$$(T)\,\frac{\alpha(t_1) \qquad t_1 = t_2}{\alpha(t_2)}$$

En efecto, si $\alpha(x) \equiv s_1(x) = s_2(t)$, podemos deducir:

(1)	$s_1(t_1) = s_2(t_1)$	Premisa $\alpha(t_1)$
(2)	$t_1 = t_2$	Premisa
(3)	$s_1(t_1) = s_1(t_1)$	S_2, 2
(4)	$s_2(t_1) = s_2(t_2)$	S_2, 2
(5)	$s_1(t_2) = s_2(t_2)$	S, T, 1, 3, 4

Definición de funtores por términos Veamos ahora que todo término puede usarse para definir un funtor:

Teorema 1.13 *Si $t(x_1, \ldots, x_n)$ es un término cuyas variables están todas entre $x_1, \ldots, x_n$, entonces existe un funtor n-ádico f_t tal que*

$$\underset{\text{ARP}}{\vdash} f_t(x_1, \ldots, x_n) = t(x_1, \ldots, x_n).$$

Demostración: Razonamos por inducción sobre la longitud de t. Si $t \equiv 0$, basta tomar $f_0 \equiv \kappa(c, p_1^n)$, pues la definición de f_0 es:

$$f_0(x_1, \ldots, x_n) = c(p_n^1(x_1, \ldots, x_n)),$$

y por otro lado, la definición de c es $c(x) = 0$, luego aplicando S_1 obtenemos

$$c(p_n^1(x_1, \ldots, x_n)) = 0,$$

y finalmente T nos da la igualdad

$$f_0(x_1, \ldots, x_n) = 0.$$

Si $t \equiv x_i$ basta tomar $f_t \equiv p_i^n$. La alternativa es que $t \equiv f(t_1, \ldots, t_m)$, para cierto funtor f y ciertos términos t_i para los que, por hipótesis de inducción, existen funtores f_{t_i} tales que

$$\underset{\text{ARP}}{\vdash} f_{t_i}(x_1, \ldots, x_n) = t_i(x_1, \ldots, x_n).$$

Basta tomar $f_t \equiv \kappa(f, f_{t_1}, \ldots, f_{t_m})$, pues la definición de f_t es

$$f_t(x_1, \ldots, x_n) = f(f_{t_1}(x_1, \ldots, x_n), \ldots, f_{t_m}(x_1, \ldots, x_n)).$$

Si llamamos $\bar{t}_i \equiv f_{t_i}(x_1, \ldots, x_n)$, tenemos $f_t(x_1, \ldots, x_n) = f(\bar{t}_1, \ldots, \bar{t}_n)$, así como $\bar{t}_i = t_i$, luego aplicando S_2 sucesivamente obtenemos

$$f(\bar{t}_1, \ldots, \bar{t}_n) = f(t_1, \bar{t}_2, \ldots, \bar{t}_n),$$

$$f(t_1, \bar{t}_2, \ldots, \bar{t}_n) = f(t_1, t_2, \bar{t}_3, \ldots, \bar{t}_n),$$

etc., y aplicando T concluimos

$$f_t(x_1, \ldots, x_n) = f(t_1, \ldots, t_n) \equiv t.$$

■

En principio, todos los funtores definidos por recursión tienen al menos rango 2, pero, con la ayuda del teorema anterior, vamos a ver ahora que también es posible definir funtores monádicos de este modo:

Teorema 1.14 *Si t es un término sin variables y $h(x,y)$ es un término con a lo sumo las dos variables indicadas, existe un funtor monádico f que cumple*

$$f(0) = t, \qquad f(Sx) = h(x, f(x)).$$

DEMOSTRACIÓN: Por el teorema anterior existe un funtor para el que se cumple $g(y) = t$ y otro que cumple $\bar{h}(x,y) = h(x,y)$. Consideramos el funtor diádico con axiomas

$$\bar{f}(y,0) = g(y), \qquad \bar{f}(y,Sx) = h(x,\bar{f}(y,x)),$$

y a su vez definimos $f(x) = \kappa(\bar{f}, c, p_1^1)$. Así $f(x) = \bar{f}(0,x)$, luego

$$f(0) = \bar{f}(0,0) = g(0) = t, \quad f(Sx) = \bar{f}(0,Sx) = \bar{h}(x,\bar{f}(0,x)) = h(x,f(x)).$$

■

Recogemos en el teorema siguiente los resultados que acabamos de demostrar, y que son la forma más natural de trabajar con los funtores de ARP:

Teorema 1.15 *Se cumple:*

1. *Si $t(x_1,\ldots,x_n)$ es un término cuyas variables están todas entre $x_1,\ldots,x_n$, entonces existe un funtor n-ádico f_t tal que*

$$f_t(x_1,\ldots,x_n) = t(x_1,\ldots,x_n).$$

2. *Si $g(x_1,\ldots,x_n)$, $h(x_1,\ldots,x_n,x,y)$ son términos con a lo sumo las variables indicadas, existe un funtor $n+1$-ádico f que cumple:*

$$f(x_1,\ldots,x_n,0) = g(x_1,\ldots,x_n),$$

$$f(x_1,\ldots,x_n,Sx) = h(x_1,\ldots,x_n,x,f(x_1,\ldots,x_n,x)).$$

3. *Si t es un término sin variables y $h(x,y)$ es un término con a lo sumo las variables indicadas, existe un funtor monádico f que cumple*

$$f(0) = t, \qquad f(Sx) = h(x,f(x)).$$

La única variante en 2) y 3) es que en los miembros derechos ponemos términos arbitrarios en vez de funtores, lo cual es lícito gracias a 1), y ello nos evita toda preocupación sobre que el número de variables sea el exigido por el esquema de recursión.

Notemos también que, en 2), el hecho de que la recursión se haga sobre el último argumento de F es anecdótico. Por ejemplo, si queremos que la recursión se haga sobre el primero basta considerar el funtor:

$$f^*(x_1,\ldots,x_n) = f(p_2^{n+1}(x_1,\ldots,x_n),\ldots,p_{n+1}^{n+1}(x_1,\ldots,x_n),p_1^n(x_1,\ldots,x_n)),$$

que cumple

$$f^*(0,x_1,\ldots,x_n) = g(x_1,\ldots,x_n),$$

$$f^*(Sx,x_1,\ldots,x_n) = h(x_1,\ldots,x_n,x,f^*(x,x_1,\ldots,x_n)).$$

Inducción Ahora enunciamos dos casos particulares de la regla de inducción:

$$(I_1)\ \frac{t(Sx) = t(x)}{t(x) = t(0)} \qquad (I_2)\ \frac{s_1(0) = s_2(0) \qquad s_1(Sx) = s_2(Sx)}{s_1(x) = s_2(x)}$$

La regla I_1 afirma que si un término toma el mismo valor en cada número natural y en su siguiente, entonces tiene que ser constante. Para probarlo aplicamos la regla de inducción a los términos $s_1(x) \equiv t(x)$, $s_2(x) \equiv t(0)$, $h(x, y) \equiv y$, con lo que se reduce a:

$$\frac{t(0) = t(0) \qquad t(Sx) = t(x) \qquad t(0) = t(0)}{t(x) = t(0)}$$

Vemos que las premisas primera y tercera son un mismo teorema y la segunda es la premisa de I_1, luego, en efecto, la conclusión se deduce dicha premisa.

La regla I_2 se sigue inmediatamente de I_0 usando $h(x, y) \equiv s_2(Sx)$, pues así la regla I_0 se reduce a:

$$\frac{s_1(0) = s_2(0) \qquad s_1(Sx) = s_2(Sx) \qquad s_2(Sx) = s_2(Sx)}{s_1(x) = s_2(x)}$$

de modo que las dos primeras premisas son las premisas de I_2 y la tercera es un teorema. ∎

Sustitución de variables En la sección anterior hemos señalado que, en general, será irrelevante qué variables concretas consideramos en cada momento. Ahora podemos precisar este hecho. Consideremos, por ejemplo, este axioma de ARP:

$$p_1^2(x_0, x_1) = x_0,$$

donde suponemos que las variables son precisamente las de índices 0 y 1. Consideremos ahora dos variables cualesquiera x, y y otras dos variables y_0, y_2 distintas de x_0, x_1, x, y. Aplicando sucesivamente la regla S_1 podemos ir deduciendo:

$$\begin{array}{ll} (1) & p_1^2(x_0, x_1) = x_0 \\ (2) & p_1^2(y_0, x_1) = y_0 \\ (3) & p_1^2(y_0, y_1) = y_0 \\ (4) & p_1^2(x, x_1) = x \\ (5) & p_1^2(x, y) = x \end{array}$$

Así pues, o bien consideramos que todas las fórmulas $p_1^2(x, y) = x$ son axiomas, o bien especificamos unas variables en concreto para los axiomas, pero en este segundo caso las fórmulas con otras elecciones de variables son teoremas y en la práctica es lo mismo. ∎

1.3 Aritmética básica en ARP

Todavía no estamos en condiciones de "usar cómodamente" ARP, pero vamos a probar algunos resultados elementales que, por una parte, nos permitirán formarnos una primera idea del funcionamiento de esta teoría axiomática y, por otra, serán necesarios para probar los resultados que nos permitirán manejarla con soltura.

La suma Empezamos recordando la definición del funtor suma. Según hemos explicado, para definir un funtor, en la práctica, basta presentar los axiomas que lo definen, que en el caso de la suma son las ecuaciones:

$$x + 0 = x, \qquad x + Sy = S(x + y).$$

Por el teorema 1.15 es inmediato que estas ecuaciones determinan, en efecto, un funtor diádico $+$. He aquí algunos teoremas de ARP relativos a la suma:

1. $Sx = x + 1$,
2. $(x + y) + z = x + (y + z)$,
3. $x + y = y + x$.

Demostración: En la práctica no es necesario que escribamos explícitamente cada demostración en ARP como una sucesión de fórmulas. Por ejemplo, para justificar que 1) es un teorema de ARP al lector le debería bastar esto:

$$x + 1 = x + S0 = S(x + 0) = Sx.$$

Una demostración explícita (teniendo en cuenta que $x + 1 \equiv s + S0$) sería así:

(1)	$x + Sy = S(x + y)$	Definición de suma
(2)	$x + 1 = S(x + 0)$	$S_1, 1$
(3)	$x + 0 = x$	Definición de suma
(4)	$S(x + 0) = Sx$	$S_2, 3$
(5)	$x + 1 = Sx$	$T, 2, 4$
(6)	$Sx = x + 1$	$S, 5$

La fórmula 2) la demostramos por inducción sobre z, es decir, considerando los términos $s_1(z) \equiv (x+y)+z$, $s_2(z) \equiv x+(y+z)$. La regla I_0 requiere probar en primer lugar que

$$s_1(0) \equiv (x + y) + 0 = x + y = x + (y + 0) \equiv s_2(0).$$

En segundo lugar tenemos que calcular

$$s_1(Sz) \equiv (x + y) + Sz = S((x + y) + z) \equiv S(s_1(z)),$$

$$s_2(z) \equiv x + (y + Sz) = x + S(y + z) = S(x + (y + z)) \equiv S(s_2(z)).$$

Vemos así que $s_1(Sz)$ y $s_2(Sz)$ se obtienen, respectivamente, de $s_1(z)$ y $s_2(z)$ aplicando un mismo funtor (en este caso S), y con esto ya podemos aplicar I_0 para concluir la igualdad.

De nuevo damos la prueba detallada para que el lector vea cómo se pueden completar fácilmente estos argumentos con sustituciones y las propiedades del igualador:

(1)	$x + 0 = x$	Definición de suma
(2)	$(x + y) + 0 = x + y$	$S_1, 1$
(3)	$y + 0 = y$	Definición de suma
(4)	$x + (y + 0) = x + y$	$S_2, 3$
(5)	$(x + y) + 0 = x + (y + 0)$	$T, 2, 4$
(6)	$x + Sz = S(x + z)$	Definición de suma
(7)	$(x + y) + Sz = S((x + y) + z)$	$S_1, 6$
(8)	$y + Sz = S(y + z)$	Definición de suma
(9)	$x + (y + Sz) = x + S(y + z)$	$S_2, 8$
(10)	$x + S(y + z) = S(x + (y + z))$	$S_1, 6$
(11)	$x + (y + Sz) = S(x + (y + z))$	$T, 9, 10$
(12)	$(x + y) + z = x + (y + z)$	$I_0, 5, 7, 11$

Para probar 3) probamos primero un caso particular: $0+x = x$. Razonamos por inducción con $s_1(x) = 0 + x$, $s_2(x) = x$. Un esbozo de la prueba consiste en observar que, por definición de suma $s_1(0) \equiv 0 + 0 = 0 \equiv s_2(0)$ y, por otra parte,

$$s_1(Sx) \equiv 0 + Sx = S(0 + x) \equiv S(s_1(x)), \quad s_2(Sx) \equiv Sx = Sx \equiv S(s_2(x)),$$

de modo que $s_1(Sx)$ y $s_2(Sx)$ se obtienen de $s_1(x)$ y $s_2(x)$ respectivamente mediante el mismo cálculo (aplicar el funtor S), luego la regla I_0 nos da la conclusión.

En segundo lugar probamos que $Sy + x = y + Sx$, ahora por inducción con $s_1(x) \equiv Sy + x$, $s_2(x) \equiv y + Sx$. En primer lugar:

$$s_2(0) \equiv y + S0 = S(y + 0) = Sy = Sy + 0 \equiv s_1(0).$$

Por otra parte,

$$s_1(Sx) \equiv Sy + Sx = S(Sy + x) \equiv S(s_1(x)),$$

$$s_2(Sx) \equiv y + SSx = S(y + Sx) \equiv S(s_2(x)),$$

y podemos aplicar la regla I_0.

Por último razonamos por inducción con $s_1(y) \equiv x + y$, $s_2(y) \equiv y + x$. Por la primera igualdad que hemos probado:

$$s_1(0) \equiv x + 0 = x = 0 + x \equiv s_2(0).$$

Y usando la segunda igualdad:

$$s_1(Sy) \equiv x + Sy = S(x + y) \equiv S(s_1(y)),$$

$$s_2(Sy) \equiv Sy + x = y + Sx = S(y + x) \equiv S(s_2(y)),$$

y también podemos aplicar la regla I_0. Dejamos a cargo del lector dar una prueba explícita. ∎

El producto Ahora recordamos la definición del producto de números naturales, que está determinado por los axiomas:

$$x \cdot 0 = 0, \qquad x \cdot Sy = x \cdot y + x.$$

De nuevo el teorema 1.15 garantiza que esta definición es correcta. Los teoremas básicos sobre el producto son:

1. $x \cdot 1 = x$,
2. $x \cdot y = y \cdot x$,
3. $x \cdot (y + z) = x \cdot y + x \cdot z$,
4. $x(yz) = (xy)z$.

DEMOSTRACIÓN: 1) $x \cdot 1 = x \cdot S0 = x \cdot 0 + x = 0 + x = x$.

2) Veamos en primer lugar que $0 \cdot x = 0$. Razonamos por inducción con $s_1(x) \equiv 0 \cdot x$, $s_2(x) \equiv 0$. Entonces

$$s_1(0) \equiv 0 \cdot 0 = 0 \equiv s_2(0).$$

Por otra parte,

$$s_1(Sx) \equiv 0 \cdot Sx = 0 \cdot x + 0 = 0 \cdot x \equiv s_1(x),$$

$$s_2(Sx) \equiv (Sx) \cdot 0 = 0 \equiv s_2(x),$$

luego la regla de inducción nos da la conclusión.

Ahora tomamos $s_1(y) \equiv (Sx) \cdot y$, $s_2(y) \equiv x \cdot y + y$. Entonces

$$s_1(0) \equiv (Sx) \cdot 0 = 0 = 0 + 0 = x \cdot 0 + 0 \equiv s_2(0).$$

Por otro lado[4]

$$s_1(Sy) \equiv (Sx) \cdot Sy = (Sx) \cdot y + Sx \equiv s_1(y) + Sx,$$

[4]Notemos que una vez probada la asociatividad de la suma ya no necesitamos distinguir entre términos como $(x \cdot y + y) + Sx$ y $x \cdot y + (y + Sx)$, pues se puede demostrar que son iguales, luego intercambiables, por lo que no hace falta poner paréntesis en las sumas de varios sumandos. Lo mismo valdrá para el producto en cuanto hayamos probado su asociatividad.

$$s_2(Sy) \equiv x \cdot Sy + Sy = x \cdot y + x + Sy = S(x \cdot y + x + y)$$
$$= S(x \cdot y + y + x) = x \cdot y + y + Sx \equiv s_2(y) + Sx.$$

La regla de inducción nos da que $(Sx) \cdot y = x \cdot y + y$.

Finalmente tomamos $s_1(x) \equiv x \cdot y$, $s_2(x) \equiv y \cdot x$. Entonces

$$s_1(0) \equiv 0 \cdot y = 0 = y \cdot 0 \equiv s_2(y).$$

$$s_1(Sx) \equiv Sx \cdot y = x \cdot y + y \equiv s_1(x) + y,$$
$$s_2(Sx) \equiv y \cdot Sx = y \cdot x + y \equiv s_2(x) + y,$$

luego concluimos que $x \cdot y = y \cdot x$.

3) Tomamos $s_1(z) \equiv x \cdot (y + z)$, $s_2(z) \equiv x \cdot y + x \cdot z$. Entonces

$$s_1(0) \equiv x \cdot (y + 0) = x \cdot y = x \cdot y + 0 = x \cdot y + x \cdot 0 \equiv s_2(0).$$

Por otra parte

$$s_1(Sz) \equiv x \cdot (y + Sz) = x \cdot S(y + z) = x \cdot (y + z) + x \equiv s_1(z) + x,$$

$$s_2(Sz) \equiv x \cdot y + x \cdot Sz = x \cdot y + x \cdot z + x \equiv s_2(z) + x,$$

luego la regla de inducción nos permite concluir. La prueba de 4) es similar. ∎

Cálculos explícitos El lector sabrá sin duda que $2 \cdot 3 = 6$, pero una cosa es que una afirmación aritmética sea verdadera y otra muy distinta que pueda demostrarse en ARP. Para que suceda lo segundo es necesario que la afirmación se pueda expresar mediante una fórmula de $\mathcal{L}_{\text{arp}}$ (lo cual es cierto en este caso, pues podemos considerar a $2 \cdot 3 = 6$ como una fórmula de $\mathcal{L}_{\text{arp}}$ que significa precisamente que "dos por tres son seis") y además que dicha fórmula pueda demostrarse a partir de los axiomas de ARP usando las reglas de inferencia de ARP, y esto no sucede con todas las afirmaciones verdaderas formalizables en ARP. No obstante, en el caso de esta fórmula en particular es fácil encontrar una demostración:

$$\begin{aligned}
2 \cdot 3 &= 2 \cdot 2 + 2 = (2 \cdot 1 + 2) + 2 = ((2 \cdot 0 + 2) + 2) + 2 = ((0 + 2) + 2) + 2 \\
&= S(((0 + 2) + 2) + 1) = SS(((0 + 2) + 2) + 0) = SS((0 + 2) + 2) \\
&= SS(S((0 + 2) + 1)) = SS(SS((0 + 2) + 0)) = SSSS(0 + 2) \\
&= SSSSS(0 + 1) = SSSSSS(0 + 0) = SSSSSS0 = 6.
\end{aligned}$$

Podríamos haber abreviado la demostración usando algunos de los teoremas que hemos demostrado sobre la suma y el producto, pero hemos dado una demostración basada exclusivamente en las definiciones de la suma y del producto (y de los numerales). Como el producto se define a partir de la suma y la suma a

partir del funtor S, lo que hemos hecho ha sido reducir el cálculo de un producto al cálculo de varias sumas, y el cálculo de las sumas a aplicar varias veces el operador S.

Es fácil convencerse de que en ARP se puede calcular cualquier suma y cualquier producto, pero en realidad sucede algo mucho más general: todas las funciones expresables mediante funtores de ARP se pueden calcular en la práctica, en el sentido de que el resultado se puede reducir a un numeral (no en vano son las funciones recursivas primitivas). Esto es lo que prueba el teorema siguiente:

Teorema 1.16 *Si t es un término sin variables y $d = N(t)$ es el número natural denotado por t, entonces $\underset{\text{ARP}}{\vdash} t = \bar{d}$, donde $\bar{d}$ es el numeral que cumple $N(\bar{d}) = d$.*

Demostración: En primer lugar demostramos el teorema en el caso en que $t \equiv f(\bar{d}_1, \ldots, \bar{d}_n)$, donde f es un funtor y cada d_i es un número natural, de modo que $d = F(f)(d_1, \ldots, d_n)$. Razonamos por inducción sobre la longitud de f.

1. Si $f \equiv S$, entonces $t \equiv Sd_1 \equiv \overline{d_1 + 1} \equiv \bar{d}$ y, ciertamente, $\underset{\text{ARP}}{\vdash} \bar{d} = \bar{d}$.

2. Si $f \equiv c$, entonces $t \equiv c(\bar{d}_1)$, luego $d = F(c)(d_1) = 0$ y, como $c(x) = \bar{0}$ es un axioma de ARP, la regla S_1 nos da $\underset{\text{ARP}}{\vdash} t = \bar{0}$.

3. Si $f \equiv p^n_k$, entonces $t \equiv p^n_k(\bar{d}_1, \ldots, \bar{d}_n)$, luego $d = F(p^n_k)(d_1, \ldots, d_n) = d_k$. Como $p^n_k(x_1, \ldots, x_n) = x_k$ es un axioma de ARP, aplicando n veces la regla S_1 obtenemos que $\underset{\text{ARP}}{\vdash} t = \bar{d}$.

4. Si $f \equiv \kappa(h, g_1, \ldots, g_m)$, tenemos entonces que $d = F(h)(e_1, \ldots, e_m)$, donde $e_i = F(g_i)(d_1, \ldots, d_n)$. Por hipótesis de inducción $\underset{\text{ARP}}{\vdash} g_i(\bar{d}_1, \ldots, \bar{d}_n) = \bar{e}_i$, así como que
$$\underset{\text{ARP}}{\vdash} h(\bar{e}_1, \ldots, \bar{e}_m) = \bar{d}.$$
Aplicando m veces la regla S_2, de aquí obtenemos
$$\underset{\text{ARP}}{\vdash} h(g_1(\bar{d}_1, \ldots, \bar{d}_n), \ldots, g_m(\bar{d}_1, \ldots \bar{d}_n)) = \bar{d},$$
y aplicando S_1 a la definición de f y la regla T concluimos $\underset{\text{ARP}}{\vdash} t = \bar{d}$.

5. Si $f \equiv \rho(g, h)$, entonces d viene determinado por las relaciones
$$e_0 = N(g)(d_1, \ldots, d_{n-1}), \quad e_{i+1} = N(h)(d_1, \ldots, d_{n-1}, i, e_i),$$
de modo que $d = e_{d_n}$. Por hipótesis de inducción
$$\underset{\text{ARP}}{\vdash} g(\bar{d}_1, \ldots, \bar{d}_{n-1}) = \bar{e}_0,$$
y teniendo en cuenta que
$$f(x_1, \ldots, x_{n-1}, \bar{0}) = g(x_1, \ldots, x_{n-1})$$

es un axioma de ARP (parte de la definición de f), aplicando S_1 llegamos a que

$$\underset{\text{ARP}}{\vdash} f(\bar{d}_1, \ldots, \bar{d}_{n-1}, \bar{0}) = g(\bar{d}_1, \ldots, \bar{d}_{n-1})$$

y por T concluimos que $\underset{\text{ARP}}{\vdash} f(\bar{d}_1, \ldots, \bar{d}_{n-1}, \bar{0}) = \bar{e}_0$.

Veamos que, en general,

$$\underset{\text{ARP}}{\vdash} f(\bar{d}_1, \ldots, \bar{d}_{n-1}, \bar{\imath}) = \bar{e}_i.$$

Lo tenemos probado para $i = 0$. Razonando por inducción sobre i, lo suponemos cierto para i y usamos que, por la hipótesis de inducción sobre la longitud de f (aplicada al funtor h):

$$\underset{\text{ARP}}{\vdash} h(\bar{d}_1, \ldots, \bar{d}_{n-1}, \bar{\imath}, \bar{e}_i) = \bar{e}_{i+1}.$$

Usando la definición de f, según la cual

$$f(x_1, \ldots, x_{n-1}, Sx) = h(x_1, \ldots, x_{n-1}, x, f(x_1, \ldots, x_{n-1}, x)),$$

aplicando S_1 obtenemos

$$\underset{\text{ARP}}{\vdash} f(\bar{d}_1, \ldots, \bar{d}_{n-1}, \overline{i+1}) = h(\bar{d}_1, \ldots, \bar{d}_{n-1}, \bar{\imath}, f(\bar{d}_1, \ldots, \bar{d}_{n-1}, \bar{\imath})).$$

La hipótesis de inducción sobre i junto con las reglas S_2 y T nos dan

$$\underset{\text{ARP}}{\vdash} f(\bar{d}_1, \ldots, \bar{d}_{n-1}, \overline{i+1}) = h(\bar{d}_1, \ldots, \bar{d}_{n-1}, \bar{\imath}, \bar{e}_i),$$

y la hipótesis de inducción sobre h, junto con T, nos permite concluir que

$$\underset{\text{ARP}}{\vdash} f(\bar{d}_1, \ldots, \bar{d}_{n-1}, \overline{i+1}) = \bar{e}_{i+1}.$$

Esto completa la inducción sobre i, con lo que, en particular, para $i = d_n$, tenemos que

$$\underset{\text{ARP}}{\vdash} f(\bar{d}_1, \ldots, \bar{d}_n) = \bar{d}.$$

Esto termina la prueba del caso particular. Para probar el caso general razonamos por inducción sobre la longitud de t. Si $t \equiv \bar{0}$, entonces $d = 0$ y ciertamente $\underset{\text{ARP}}{\vdash} \bar{0} = \bar{0}$.

La alternativa es que $f \equiv f(t_1, \ldots, t_n)$, para cierto funtor f y ciertos términos sin variables t_i que, por hipótesis de inducción, llamando $d_i = N(t_i)$, cumplirán

$$\underset{\text{ARP}}{\vdash} t_i = \bar{d}_i.$$

Además, $d = N(t) = F(f)(d_1, \ldots, d_n)$. Por la parte ya probada,

$$\underset{\text{ARP}}{\vdash} f(\bar{d}_1, \ldots, \bar{d}_n) = \bar{d},$$

luego aplicando n veces la regla S_2 y T llegamos a que $\underset{\text{ARP}}{\vdash} t = \bar{d}$. ∎

Así pues, si comprobamos que una función recursiva primitiva, sobre unos valores dados, toma un cierto valor, independientemente de cómo hayamos llegado a esa conclusión, podemos asegurar que ese hecho puede probarse razonando en ARP (simplemente, aplicando las definiciones de la función considerada y las que se usan para definirla).

La resta truncada Consideramos ahora los funtores cuyos axiomas son

$$\mathrm{pre}(0) = 0, \quad \mathrm{pre}(Sx) = x,$$

$$x \dot{-} 0 = x, \qquad x \dot{-} Sy = \mathrm{pre}(x \dot{-} y).$$

Estos funtores los discutimos ya en los ejemplos de la página 8. Allí vimos que las funciones que denotan son

$$\mathrm{pre}(n) = \begin{cases} 0 & \text{si } n = 0, \\ n-1 & \text{si } n > 0, \end{cases} \qquad m \dot{-} n = \begin{cases} m-n & \text{si } m \geq n, \\ 0 & \text{si } m \leq n. \end{cases}$$

Todo esto es evidente y, sin embargo, no es fácil expresar estos hechos en el lenguaje de ARP. ¿Qué fórmula expresa que $x \dot{-} y = 0$ cuando $x \leq y$? Incluso, restringiéndonos a lo que sabemos expresar en ARP, de las premisas $x \dot{-} y = 0$, $y \dot{-} x = 0$, tendría que poder deducirse que $x = y$ y, en efecto, se puede, pero ¿cómo? Veremos que la prueba es complicada, y el hecho de que algo tan evidente sea difícil de probar da una idea de que ARP no es, tal y como la conocemos, una teoría muy "amigable".

Empezamos probando algunos resultados sencillos:

1. $x \dot{-} 1 = \mathrm{pre}\, x$.

   En efecto: $x \dot{-} 1 = \mathrm{pre}(x \dot{-} 0) = \mathrm{pre}\, x$.

2. $Sx \dot{-} Sy = x \dot{-} y$.

   Razonamos por inducción sobre y:

   $$Sx \dot{-} S0 = \mathrm{pre}(Sx \dot{-} 0) = \mathrm{pre}\, Sx = x = x \dot{-} 0,$$

   $$Sx \dot{-} SSy = \mathrm{pre}(Sx \dot{-} Sy), \qquad x \dot{-} Sy = \mathrm{pre}(x \dot{-} y),$$

   luego la regla de inducción nos da la igualdad.

3. $x \dot{-} x = 0$.

   Podemos aplicar la variante I_1 a $t(x) \equiv x \dot{-} x$. Como $Sx \dot{-} Sx = x \dot{-} x$, la conclusión es $x \dot{-} x = 0 \dot{-} 0 = 0$.

4. $0 \dot{-} x = 0$.

   Razonamos por inducción con $s_1(x) \equiv 0 \dot{-} x$ y $s_2(x) \equiv 0$. Así

   $$s_1(0) = 0 \dot{-} 0 = 0 = s_2(0),$$

   $$s_1(Sx) = 0 \dot{-} Sx = \mathrm{pre}(0 \dot{-} x) = \mathrm{pre}\, s_1(x),$$

   $$s_2(Sx) = 0 = \mathrm{pre}\, 0 = \mathrm{pre}\, s_2(x).$$

5. $(x + y) \dot{-} y = x$.

   Aplicamos I_1 a $t(y) \equiv (x + y) \dot{-} y$. Se cumple que

   $$t(Sy) = (x + Sy) \dot{-} Sy = S(x + y) \dot{-} Sy = (x + y) \dot{-} y = t(y),$$

   luego la conclusión es $(x + y) \dot{-} y = (x + 0) \dot{-} 0 = x$.

6. $(x + y) \dot{-} x = y$.

   Esto es consecuencia del resultado precedente y de la conmutatividad de la suma.

7. $(x + z) \dot{-} (y + z) = x \dot{-} y$.

   Aplicamos I_1 a $t(z) \equiv (x + z) \dot{-} (y + z)$.

   $$t(Sz) = (x + Sz) \dot{-} (y + Sz) = S(x + z) \dot{-} S(y + z) = (x + z) \dot{-} (y + z),$$

   luego $(x + z) \dot{-} (y + z) = (x + 0) \dot{-} (y + 0) = x \dot{-} y$.

8. $y \dot{-} (x + y) = 0$.

   En efecto, por 7,

   $$y \dot{-} (x + y) = (0 + y) \dot{-} (x + y) = 0 \dot{-} x = 0.$$

9. $x \cdot (1 \dot{-} x) = 0$.

   Por la regla I_2:

   $$0 \cdot (1 \dot{-} 0) = 0, \qquad Sx(1 \dot{-} Sx) = Sx(0 \dot{-} x) = Sx \cdot 0 = 0.$$

10. $(1 \dot{-} x)y = y \dot{-} xy$.

    Por inducción respecto de x:

    $$(1 \dot{-} 0)y = y = y \dot{-} 0 = y \dot{-} (0 \cdot y),$$

    $$(1 \dot{-} Sx)y = (0 - x)y = 0 \cdot y = 0,$$

    $$y \dot{-} (Sx) \cdot y = y \dot{-} (y + xy) = 0,$$

    donde en el último paso hemos usado 8.

Ahora abordamos el problema de deducir $x = y$ de las premisas $x \dot{-} y = 0$, $y \dot{-} x = 0$. No tenemos una forma de llevar a cabo una inducción que tenga en cuenta unas premisas. En su lugar, vamos a probar una igualdad válida sin premisas, pero que, al suponer $x \dot{-} y = 0$, $y \dot{-} x = 0$, se reduce a $x = y$:

Teorema 1.17 $x + (y \dot{-} x) = y + (x \dot{-} y)$.

DEMOSTRACIÓN: Sea $t_1(x,y) \equiv x + (y \dot{-} x)$, de modo que

$$t_1(x,0) = x, \quad t_1(0,y) = y, \qquad t_1(Sx,Sy) = St_1(x,y).$$

Igualmente, sea $t_2(x,y) \equiv y + (x \dot{-} y)$, de modo que

$$t_2(x,0) = x, \quad t_2(0,y) = y, \quad t_2(Sx,Sy) = St_2(x,y).$$

En primer lugar demostramos que

$$(x \dot{-} 1) + (1 \dot{-} (1 \dot{-} x)) = x.$$

Aplicamos la regla I_2. Para $x = 0$ se cumple la igualdad, pues

$$(0 \dot{-} 1) + (1 \dot{-} (1 \dot{-} 0)) = 0 + (1 \dot{-} 1) = 0.$$

Por otro lado, usando la propiedad 2 de la resta truncada,

$$(Sx \dot{-} 1) + (1 \dot{-} (1 \dot{-} Sx)) = x + (1 \dot{-} (0 \dot{-} x)) = x + 1 = Sx,$$

luego I_2 nos permite concluir.

De aquí deducimos a su vez que

$$t_1(x,y) = t_1(x \dot{-} 1, y \dot{-} 1) + (1 \dot{-} (1 \dot{-} (x + y))). \tag{1.1}$$

Aplicamos de nuevo la regla I_2. El miembro derecho en $y = 0$ es

$$t_1(x \dot{-} 1, 0) + (1 \dot{-} (1 \dot{-} x)) = x \dot{-} 1 + (1 \dot{-} (1 \dot{-} x)) = x = t_1(x,0),$$

por la igualdad que hemos demostrado previamente. Por otra parte,

$$t_1(x \dot{-} 1, Sy \dot{-} 1) + (1 \dot{-} (1 \dot{-} (x + Sy))) =$$

$$t_1(x \dot{-} 1, y) + (1 \dot{-} (1 \dot{-} S(x + y))) =$$

$$t_1(x \dot{-} 1, y) + (1 \dot{-} (0 \dot{-} (x + y))) =$$

$$t_1(x \dot{-} 1, y) + 1 = St_1(x \dot{-} 1, y).$$

Por lo tanto, para aplicar I_2 basta demostrar la igualdad

$$t_1(x, Sy) = St_1(x \dot{-} 1, y).$$

Ésta la probamos a su vez aplicando I_2. Para $x = 0$ se reduce a $Sy = Sy$. Además,

$$t_1(Sx, Sy) = St_1(x,y) = St_1(\operatorname{pre} Sx, y) = St_1(Sx \dot{-} 1, y).$$

Esto termina la prueba de (1.1). Ahora consideramos el funtor dado por las ecuaciones

$$F(x,y,0) = 0, \quad F(x,y,Sz) = F(x,y,z) + (1 \dot{-} (1 \dot{-} ((x \dot{-} z) + (y \dot{-} z)))).$$

Vamos a demostrar que

$$t_1(x \dot{-} z, y \dot{-} z) + F(x, y, z) = t_1(x \dot{-} Sz, y \dot{-} Sz) + F(x, y, Sz). \tag{1.2}$$

En efecto, por (1.1) tenemos que

$$t_1(x \dot{-} z, y \dot{-} z) + F(x, y, z) =$$

$$t_1((x \dot{-} z) \dot{-} 1, (y \dot{-} z) \dot{-} 1) + F(x, y, z) + (1 \dot{-} (1 \dot{-} ((x \dot{-} z) + (y \dot{-} z)))) =$$

$$t_1(x \dot{-} Sz, y \dot{-} Sz) + F(x, y, Sz).$$

Notemos que el miembro derecho de (1.2) resulta de sustituir z por Sz en el miembro izquierdo, luego la regla I_1 nos da que

$$t_1(x \dot{-} z, y \dot{-} z) + F(x, y, z) = t_1(x \dot{-} 0, y \dot{-} 0) + F(x, y, 0) = t_1(x, y).$$

Ahora bien, intercambiando los papeles de x e y, el mismo razonamiento vale para t_2, y nos lleva hasta

$$t_2(x \dot{-} z, y \dot{-} z) + F(x, y, z) = t_2(x, y).$$

(Notemos que la definición de F es simétrica.) Sustituyendo z por y queda

$$t_1(x \dot{-} y, 0) + F(x, y, y) = t_1(x, y), \qquad t_2(x \dot{-} y, 0) + F(x, y, y) = t_2(x, y)$$

o también:

$$t_1(x, y) = x \dot{-} y + F(x, y, y), \qquad t_2(x, y) = x \dot{-} y + F(x, y, y),$$

luego $t_1(x, y) = t_2(x, y)$, que es lo que queríamos probar. ∎

Ahora ya es inmediato que de $x \dot{-} y = 0$, $y \dot{-} x = 0$ se deduce $x = y$, pero vamos a enunciarlo en términos más convenientes:

El valor absoluto Consideramos el funtor dado por

$$|x - y| = (x \dot{-} y) + (y \dot{-} x).$$

El hecho fundamental que vamos a necesitar es que se cumplen estas dos reglas de inferencia:

$$\frac{s = t}{|s - t| = 0} \qquad \frac{|s - t| = 0}{s = t}$$

donde s, t son términos cualesquiera.

La primera es inmediata, mientras que la segunda es consecuencia del teorema anterior. En efecto, si suponemos $|s - t| = 0$, tenemos que

$$|s - t| \dot{-} (t \dot{-} s) = 0$$

(porque $0 \dot{-} x = 0$), pero esto es

$$((s \dot{-} t) + (t \dot{-} s)) - (t \dot{-} s) = 0$$

y la propiedad 5 de la resta truncada nos da que $s \dot{-} t = 0$. Análogamente (usando la propiedad 6) concluimos que $t \dot{-} s = 0$, y el teorema 1.17 nos da entonces que $s = t$.

Algunas propiedades elementales del valor absoluto son:

1. $|x - y| = |y - x|$,
2. $|x - 0| = x$,
3. $|(x + z) - (y + z)| = |x - y|$,
4. $(1 \dot{-} z)|x - y| = |(1 \dot{-} z)x - (1 \dot{-} z)y|$.

Las tres primeras son inmediatas a partir de la definición y de las propiedades de la resta truncada, y la última se prueba inmediatamente aplicando la regla I_2 sobre z.

1.4 El cálculo proposicional

En principio, las fórmulas de ARP no pueden expresar más que igualdades entre funciones recursivas primitivas, pero esto es engañoso, como veremos enseguida. La clave está en que si $\alpha \equiv s_1 = s_2$ es una fórmula arbitraria, podemos definir el término

$$t_\alpha \equiv |s_1 - s_2|,$$

de modo que las reglas de inferencia del valor absoluto se pueden enunciar así:

$$\frac{\alpha}{t_\alpha = 0} \qquad \frac{t_\alpha = 0}{\alpha} \tag{1.3}$$

Esto significa que toda fórmula es equivalente a otra de la forma $t = 0$. Notemos que si la fórmula ya es de la forma $\alpha \equiv t = 0$, entonces $t_\alpha = t$.

Esto da mucho juego, como mostramos a continuación:

Los conectores lógicos Si α y β son fórmulas, definimos:

$$\neg\alpha \equiv 1 \dot{-} t_\alpha = 0, \qquad \alpha \to \beta \equiv (1 \dot{-} t_\alpha) \cdot t_\beta = 0.$$

$$\alpha \vee \beta \equiv t_\alpha \cdot t_\beta = 0, \qquad \alpha \wedge \beta \equiv t_\alpha + t_\beta = 0, \qquad \alpha \leftrightarrow \beta \equiv (\alpha \to \beta) \wedge (\beta \to \alpha).$$

Escribiremos también $s \neq t \equiv \neg(s = t)$.

El significado de estas fórmulas es claro. Fijada una valoración v, tenemos:[5]

[5] Observemos que, si $\alpha \equiv s_1 = s_2$, se cumple que $\vDash \alpha[v]$ equivale a $|N(s_1)[v] - N(s_2)[v]| = 0$, luego a $N(t_\alpha)[v] = 0$.

1. $\vDash \alpha[v]$ si y sólo si $1 \dot{-} N(t_\alpha)[v] = 0$, si y sólo si $N(t_\alpha)[v] \neq 0$, si y sólo si no se cumple $\vDash \alpha[v]$.
2. $\vDash (\alpha \vee \beta)[v]$ si y sólo si $N(t_\alpha)[v] \cdot N(t_\beta)[v] = 0$, si y sólo si $N(t_\alpha)[v] = 0$ o $N(t_\beta)[v] = 0$, si y sólo si $\vDash \alpha[v]$ o $\vDash \beta[v]$.
3. $\vDash (\alpha \wedge \beta)[v]$ si y sólo si $N(t_\alpha)[v] + N(t_\beta)[v] = 0$, si y sólo si $N(t_\alpha)[v] = 0$ y $N(t_\beta)[v] = 0$, si y sólo si $\vDash \alpha[v]$ y $\vDash \beta[v]$.
4. $\vDash (\alpha \to \beta)[v]$ si y sólo si $(1 \dot{-} N(t_\alpha)[v]) \cdot N(t_\beta)[v] = 0$, lo cual equivale a que $1 \dot{-} N(t_\alpha)[v] = 0$ o $N(t_\beta)[v] = 0$, que a su vez equivale a que $N(t_\alpha) \neq 0$ o $N(t_\beta)[v] = 0$, que a su vez equivale a que, o bien no $\vDash \alpha[v]$, o bien $\vDash \beta[v]$.
5. Claramente $\vDash (\alpha \leftrightarrow \beta)[v]$ si y sólo α y β son equivalentes, en el sentido de que se cumple $\vDash \alpha[v]$ si y sólo si se cumple $\vDash \beta[v]$.

Con esto hemos demostrado que la fórmula $\neg\alpha$ significa "no α", que $\alpha \vee \beta$ significa "α o β", que $\alpha \wedge \beta$ significa "α y β", que $\alpha \to \beta$ significa "si se cumple α, entonces también se cumple β" y que $\alpha \leftrightarrow \beta$ significa que "α se cumple si y sólo si se cumple β".

Enseguida demostraremos que en ARP se pueden demostrar los hechos que se debería poder demostrar teniendo en cuenta esta interpretación, pero antes probaremos algunos resultados sencillos que no podríamos enunciar sin los conectores que acabamos de introducir:

Los axiomas de Peano Las fórmulas siguientes se conocen habitualmente como *axiomas de Peano.* Las últimas ya las conocíamos, pues no son más que las definiciones de la suma y del producto:

Teorema 1.18 (Axiomas de Peano) *Se cumple:*

1. $Sx \neq 0$,
2. $Sx = Sy \to x = y$,
3. $x + 0 = x$,
4. $x + Sy = S(x + y)$,
5. $x \cdot 0 = 0$,
6. $x \cdot Sy = x \cdot y + x$.

DEMOSTRACIÓN: La fórmula $Sx \neq 0$ es $1 \dot{-} |Sx - 0| = 0$, pero

$$|Sx - 0| = (Sx \dot{-} 0) + (0 \dot{-} Sx) = Sx,$$

luego tenemos que probar $1 \dot{-} Sx = 0$ y, por las propiedades 2 y 4 de la resta truncada, es $S0 \dot{-} Sx = 0 \dot{-} x = 0$.

La segunda fórmula es $(1 \dot{-} |Sx - Sy|)|x - y| = 0$, pero, usando de nuevo la propiedad 2 de la resta truncada, equivale a $(1 \dot{-} |x - y|)|x - y| = 0$, lo cual se sigue de la propiedad 9. ∎

La regla de inducción Hay un último axioma de Peano que no es sino el principio de inducción usual. Éste no puede ser expresado mediante una fórmula de $\mathcal{L}_{\mathrm{arp}}$, pero sí como una regla de inferencia:

$$(I)\ \frac{\alpha(0) \qquad \alpha(x) \to \alpha(Sx)}{\alpha(x)}.$$

DEMOSTRACIÓN: Pongamos que $\alpha(x) \equiv s(x) = t(x)$ y consideremos el funtor dado por $f(x) = t_\alpha \equiv |s(x) - t(x)|$. Claramente, basta probar la validez de la regla

$$\frac{f(0) = 0 \qquad (1 \dot{-} f(x)) \cdot f(Sx) = 0}{f(x) = 0}.$$

Para ello consideramos el funtor dado por las ecuaciones

$$g(0) = 1, \qquad g(Sx) = g(x)(1 \dot{-} f(x)).$$

Notemos que la función $F(g)(n)$ toma el valor 1 mientras $F(f)(n)$ valga 0, y pasa a valer 0 a partir del primer natural en el que $F(f)(n) \neq 0$. Por lo tanto, hay que probar de $g(x) = 1$.

Usando la propiedad 10 de la resta truncada:

$$g(SSx)) = g(Sx)(1 \dot{-} f(Sx)) = g(x)(1 \dot{-} f(x))(1 \dot{-} f(Sx))$$

$$= g(x)((1 \dot{-} f(x)) \dot{-} (1 \dot{-} f(x))f(Sx)) = g(Sx),$$

donde hemos usado la premisa $(1 \dot{-} f(x))f(Sx) = 0$. Por I_1 aplicada al término $g(Sx)$ concluimos que $g(Sx) = g(S0) = g(0)(1 \dot{-} f(0)) = 1$, donde hemos usado la premisa $f(0) = 0$.

Explícitamente, tenemos que $g(Sx) = g(x)(1 \dot{-} f(x)) = 1$, luego, multiplicando por $f(x)$, queda

$$f(x) = g(x)(1 \dot{-} f(x))f(x) = 0,$$

donde hemos usado que $x(1 \dot{-} x) = 0$. ∎

De momento, poco partido le podemos sacar a esta regla, ya que no es fácil demostrar implicaciones. Por ejemplo, todos los teoremas siguientes son trivialmente verdaderos, pero sus demostraciones no son obvias:

1. $((\alpha \to \beta) \wedge (\beta \to \gamma)) \to (\alpha \to \gamma)$.
2. $\neg\neg\alpha \to \alpha, \qquad \alpha \to \neg\neg\alpha$.
3. $((\alpha \to \beta) \wedge \neg\beta) \to \neg\alpha$.
4. $\alpha \wedge \neg\alpha \to \beta$.
5. $\alpha \vee \neg\alpha$.
6. $\alpha \to (\alpha \vee \beta), \qquad \beta \to (\alpha \vee \beta)$.

7. $(\alpha \vee \alpha) \to \alpha$.
8. $((\alpha \to \gamma) \wedge (\beta \to \gamma)) \to ((\alpha \vee \beta) \to \gamma)$.
9. $(\alpha \wedge \beta) \to \alpha, \qquad (\alpha \wedge \beta) \to \beta$.
10. $\neg(\alpha \wedge \neg\alpha)$.
11. $((\alpha \wedge \beta) \to \gamma) \to (\alpha \to (\beta \to \gamma))$.

DEMOSTRACIÓN: De momento, una estrategia que podemos emplear para probar este tipo de fórmulas con los escasos medios de que disponemos es empezar eliminando las definiciones de los conectores y luego tratar de probar (normalmente por inducción) la fórmula resultante. Por ejemplo, la fórmula 1) es

$$(1 \dot{-} t_{(\alpha\to\beta)\wedge(\beta\to\gamma)}) \cdot t_{\alpha\to\gamma} = 0,$$

pero a su vez tenemos que $t_{(\alpha\to\beta)\wedge(\beta\to\gamma)} = t_{\alpha\to\beta} + t_{\beta\to\gamma}$ y, si vamos sustituyendo las definiciones sucesivamente, llegamos a la fórmula

$$(1 \dot{-} ((1 \dot{-} t_\alpha) \cdot t_\beta + (1 \dot{-} t_\beta) \cdot t_\gamma)) \cdot (1 \dot{-} t_\alpha) \cdot t_\gamma = 0.$$

Recíprocamente, si demostramos esta fórmula, realizando las sustituciones de términos que proporcionan las definiciones de los conectores, llegamos a la fórmula 1). Para probar esta fórmula demostraremos, más en general, la fórmula

$$(1 \dot{-} ((1 \dot{-} x) \cdot y + (1 \dot{-} y) \cdot z)) \cdot (1 \dot{-} x) \cdot z = 0,$$

pues sustituyendo en ella las variables por los términos adecuados se llega a la fórmula que queremos probar. Aplicamos I_2 a la variable x.

1. Para $x = 0$ tenemos que probar

   $$(1 \dot{-} (y + (1 \dot{-} y) \cdot z)) \cdot z = 0.$$

   Demostraremos esto mediante I_2 aplicado a la variable y:

   (a) Para $y = 0$ tenemos que probar

   $$(1 \dot{-} z) \cdot z = 0,$$

   lo cual es la propiedad 9 de la resta truncada.

   (b) Para Sy tenemos que probar

   $$(1 \dot{-} (Sy + (1 \dot{-} Sy) \cdot z)) \cdot z = 0,$$

   pero el miembro izquierdo es $(1 \dot{-} Sy) \cdot z = 0 \cdot z = 0$.

2. Para Sx tenemos que probar

   $$(1 \dot{-} ((1 \dot{-} Sx) \cdot y + (1 \dot{-} y) \cdot z)) \cdot (1 \dot{-} Sx) \cdot z = 0,$$

   lo cual se cumple porque $1 \dot{-} Sx = 0$.

2) Sustituyendo t_α por una variable arbitraria, tenemos que probar las fórmulas

$$(1 \dot{-} (1 \dot{-} (1 \dot{-} x))) \cdot x = 0, \qquad (1 \dot{-} x) \cdot (1 \dot{-} (1 \dot{-} x))) = 0.$$

Las dos se prueban trivialmente mediante I_2.

3) Basta probar la fórmula

$$(1 \dot{-} ((1 \dot{-} x) \cdot y + (1 \dot{-} y))) \cdot (1 \dot{-} x) = 0.$$

Es inmediato que al sustituir x por Sx en el miembro izquierdo queda 0, luego, por I_2, basta probar que

$$1 \dot{-} (y + (1 \dot{-} y)) = 0,$$

lo cual se prueba trivialmente mediante I_2.

4) Basta probar la fórmula

$$(1 \dot{-} (x + (1 \dot{-} x))) \cdot y = 0,$$

pero el primer factor es nulo, como ya hemos señalado en el apartado precedente.

5) Es consecuencia de la igualdad $x \cdot (1 \dot{-} x) = 0$, que es la propiedad 4 de la resta truncada.

6) Basta probar $(1 \dot{-} x) \cdot x \cdot y = 0$, pero esto se cumple porque $(1 \dot{-} x) \cdot x = 0$.

7) Basta probar $(1 \dot{-} (x \cdot x)) \cdot x = 0$, lo cual se obtiene fácilmente mediante la regla I_2.

8) Basta probar

$$(1 \dot{-} ((1 \dot{-} x) \cdot z + (1 \dot{-} y) \cdot z)) \cdot (1 \dot{-} xy) \cdot z = 0.$$

Aplicamos I_2 respecto de la variable x:

1. Para $x = 0$ tenemos que probar:

$$(1 \dot{-} (z + (1 \dot{-} y) \cdot z)) \cdot z = 0.$$

Aplicamos para ello I_2 respecto de la variable y:

(a) Para $y = 0$ tenemos que probar:

$$(1 \dot{-} (z + z)) \cdot z = 0.$$

Aplicamos I_2 respecto de la variable z:

i. Es inmediato que el miembro izquierdo vale 0 en $z = 0$.

ii. Para Sz queda:

$$(1 \dot{-} (Sz + Sz)) \cdot Sz = 0,$$

lo cual se cumple porque $Sz + Sz = S(Sz + z)$.

(b) Para Sy queda:

$$(1 \dot{-} z) \cdot z = 0,$$

y esto ya lo tenemos demostrado.

2. Para Sx queda:

$$(1 \dot{-} (1 \dot{-} y) \cdot z) \cdot (1 \dot{-} (Sx) \cdot y) \cdot z = 0.$$

Lo probamos mediante I_2 respecto de la variable y:

(a) Para $y = 0$ la ecuación se reduce a $(1 \dot{-} z) \cdot z = 0$.

(b) Para Sy queda:

$$(1 \dot{-} (Sx) \cdot Sy) \cdot z = 0,$$

y esto se cumple porque $(Sx) \cdot (Sy) = (x+1) \cdot (y+1) = S(xy+x+y)$.

9) Basta probar $(1 \dot{-} (x + y)) \cdot x = 0$, lo cual se obtiene inmediatamente mediante I_2.

10) Basta probar $1 \dot{-} (x + (1 \dot{-} x)) = 0$, pero ésta se obtiene inmediatamente con I_2.

11 Basta probar

$$(1 \dot{-} ((1 \dot{-} (x + y)) \cdot z)) \cdot (1 - x) \cdot (1 - y) \cdot z = 0.$$

Vamos a aplicar la regla I_2. Es claro que al sustituir x por Sx el miembro izquierdo da 0, luego basta probar:

$$(1 \dot{-} ((1 \dot{-} y) \cdot z)) \cdot (1 - y) \cdot z = 0.$$

Probamos esto mediante I_2, y de nuevo es claro que el miembro izquierdo vale 0 si cambiamos y por Sy, y en $y = 0$ se reduce a $(1 \dot{-} z) \cdot z = 0$. ∎

El teorema siguiente es también obviamente verdadero (no es sino una reformulación de la regla S_2), pero su prueba es aún menos trivial que las anteriores:

Teorema 1.19 $x = y \to s(x) = s(y)$.

Demostración: De la regla I_2 (respecto de z) se sigue inmediatamente la fórmula

$$(1 \dot{-} z)s(y + z) = (1 \dot{-} z)s(y).$$

En particular

$$(1 \dot{-} (x \dot{-} y))s(y + (x \dot{-} y)) = (1 \dot{-} (x \dot{-} y))s(y).$$

Multiplicamos ambos miembros por $1 \dot{-} |x - y|$:

$$(1 \dot{-} |x - y|)(1 \dot{-} (x \dot{-} y))s(y + (x \dot{-} y)) = (1 \dot{-} |x - y|)(1 \dot{-} (x \dot{-} y))s(y).$$

Ahora observamos que

$$(1 \dot{-} |x - y|)(1 \dot{-} (x \dot{-} y)) = 1 \dot{-} |x - y|. \tag{1.4}$$

En efecto, por la propiedad 10 de la resta truncada:

$$(1 \dot{-} |x - y|)(1 \dot{-} (x \dot{-} y)) = (1 \dot{-} |x - y|) \dot{-} (1 \dot{-} |x - y|)(x \dot{-} y).$$

La fórmula $(1 \dot{-} w)w = 0$ nos da $(1 \dot{-} |x - y|)|x - y| = 0$, que es lo mismo que

$$(1 \dot{-} |x - y|)(x \dot{-} y) + (1 - |x - y|)(y \dot{-} x) = 0,$$

pero la propiedad 5, es decir, $(x + y) \dot{-} y = x$, implica que si una suma es 0, el primer sumando es 0, luego

$$(1 - |x - y|)(x \dot{-} y) = 0,$$

lo que prueba (1.4), y a su vez esto reduce la fórmula anterior a (1.4) hasta

$$(1 \dot{-} |x - y|)s(y + (x \dot{-} y)) = (1 \dot{-} |x - y|)s(y).$$

Por simetría, también podemos demostrar

$$(1 \dot{-} |x - y|)s(x + (y \dot{-} x)) = (1 \dot{-} |x - y|)s(x),$$

y el teorema 1.17 nos da que ambos miembros izquierdos son iguales, luego

$$(1 \dot{-} |x - y|)s(x) = (1 \dot{-} |x - y|)s(y). \tag{1.5}$$

Por consiguiente:

$$|(1 \dot{-} |x - y|)s(x) - (1 \dot{-} |x - y|)s(y)| = 0,$$

Pero la propiedad 4 del valor absoluto nos da ahora que

$$(1 \dot{-} |x - y|)|s(x) - s(y)| = 0,$$

y esto es, por definición, $x = y \to s(x) = s(y)$. ∎

Los teoremas precedentes muestran que es mucho más fácil convencerse de que una fórmula de ARP es verdadera que demostrarla en ARP. Nuestro propósito es dotarnos de las herramientas suficientes para que esto deje de ser así. En primer lugar transformamos en reglas de inferencia los resultados que hemos obtenido:

Modus ponendo ponens Esta regla nos permite "extraer" la información contenida en una implicación:

$$\text{(MP)}\ \frac{\alpha \qquad \alpha \to \beta}{\beta}$$

En efecto, por (1.3), de las premisas se deduce $t_\alpha = 0$ y $(1 \dot{-} t_\alpha) \cdot t_\beta = 0$, pero entonces

$$t_\beta = 1 \cdot t_\beta = (1 \dot{-} 0) \cdot t_\beta = (1 \dot{-} t_\alpha) \cdot t_\beta = 0,$$

luego, de nuevo por (1.3), concluimos β. Esta regla, junto con el teorema de deducción que probamos un poco más abajo, bastan para manipular las implicaciones en cualquier deducción.

Doble negación y Tertium non datur Aplicando 2 y MP obtenemos:

$$\text{(DN)}\ \frac{\alpha}{\neg\neg\alpha} \qquad \frac{\neg\neg\alpha}{\alpha}$$

La regla del *Tertium non datur,* o *Tercio excluso,* es 5, es decir,

$$\alpha \vee \neg\alpha.$$

Estas reglas, junto con el método de reducción al absurdo que probaremos un poco más abajo, bastan para manipular las negaciones en cualquier deducción.

Introducción y eliminación del conjuntor Estas dos reglas bastan para manipular conjunciones:

$$\text{(IC)}\ \frac{\alpha \qquad \beta}{\alpha \wedge \beta}, \qquad \text{(EC)}\ \frac{\alpha \wedge \beta}{\alpha}, \qquad \frac{\alpha \wedge \beta}{\beta}$$

Si suponemos α, β, tenemos $t_\alpha = 0$ y $t_\beta = 0$, luego también $t_\alpha + t_\beta = 0$, y esto es $\alpha \wedge \beta$. Las reglas de eliminación del conjuntor se siguen de la propiedad 9 y de MP.

Introducción del disyuntor y dilema Estas reglas bastan para manipular las disyunciones en una demostración:

$$\text{(ID)}\ \frac{\alpha}{\alpha \vee \beta}, \qquad \frac{\beta}{\alpha \vee \beta}, \qquad \text{(Dil)}\ \frac{\alpha \to \gamma \qquad \beta \to \gamma}{\alpha \vee \beta \to \gamma}$$

Se siguen de 6 y 8 usando MP e IC.

Introducción y eliminación del bicondicionador Son casos particulares de IC, EC, y bastan para manipular las coimplicaciones en una deducción:

$$\text{(IB)}\ \frac{\alpha \to \beta \qquad \beta \to \alpha}{\alpha \leftrightarrow \beta}, \qquad \text{(EB)}\ \frac{\alpha \leftrightarrow \beta}{\alpha \to \beta}, \qquad \frac{\alpha \leftrightarrow \beta}{\beta \to \alpha}$$

Necesitamos dos reglas de inferencia más para demostrar a continuación el resultado fundamental:

Silogismo hipotético y Modus tollendo tollens

$$\text{(SH)}\ \frac{\alpha \to \beta \qquad \beta \to \gamma}{\alpha \to \gamma}, \qquad \text{(MT)}\ \frac{\alpha \to \beta \qquad \neg\beta}{\neg\alpha}$$

Se siguen de 1 y de 3.

Finalmente podemos demostrar el resultado fundamental que nos permitirá manejar con naturalidad los conectores lógicos:

Teorema 1.20 (Teorema de deducción) *Si se cumple*

$$\alpha_1, \ldots, \alpha_n, \alpha \underset{\text{ARP}}{\vdash} \beta$$

y en la deducción no se usan las reglas S_1 o I_0 respecto de variables de α, entonces

$$\alpha_1, \ldots, \alpha_n \underset{\text{ARP}}{\vdash} \alpha \to \beta.$$

Éste es el resultado fundamental para demostrar implicaciones. Afirma que, para deducir de unas premisas una implicación $\alpha \to \beta$, podemos añadir α a las premisas y deducir β. Vamos a ver que a partir de la deducción de β (siempre y cuando cumpla las restricciones del enunciado) se puede construir una deducción de $\alpha \to \beta$. Pero antes de eso vamos a ver un uso típico del teorema de deducción. Vamos a probar que

$$\underset{\text{ARP}}{\vdash} (\neg\alpha \to \neg\beta) \to (\beta \to \alpha).$$

En la práctica, la mejor forma de expresar la prueba es la siguiente:

|	(1)	$\neg\alpha \to \neg\beta$	Hipótesis
|	(2)	β	Hipótesis
|	(3)	$\neg\neg\beta$	DN 2
|	(4)	$\neg\neg\alpha$	MT 1, 3
|	(5)	α	DN 4
|	(6)	$\beta \to \alpha$	
	(7)	$(\neg\alpha \to \neg\beta) \to (\beta \to \alpha)$	

En la línea (1) indicamos que la fórmula es una hipótesis, es decir, una premisa provisional, añadida para aplicar el teorema de deducción. Lo mismo sucede en la línea (2), donde introducimos β como hipótesis para una segunda aplicación del teorema de deducción. Al llegar a la línea (6) aplicamos el teorema de deducción para concluir $\beta \to \alpha$ y marcamos las líneas desde la (2) hasta la (5) con una raya vertical para indicar que ninguna de ellas puede usarse en lo sucesivo, ya que todas ellas suponen la hipótesis β, con la que ya no contamos. En la línea (7) aplicamos de nuevo el teorema de deducción y marcamos las líneas desde la (1) hasta la (6) para indicar que ninguna de ellas podría usarse en lo sucesivo (si no hubiéramos terminado ya la demostración) porque suponen la hipótesis (1).

En esta demostración no hemos usado en ningún momento las reglas S_1 o I_0, por lo que no hemos tenido que preocuparnos por las restricciones que impone el teorema de deducción sobre su uso. No obstante, para que encadenar varios usos del teorema de deducción tal y como hemos hecho esté justificado, necesitamos probar una ligera variante. Concretamente, demostraremos esta versión del teorema:

A partir de una deducción

$$\alpha_1, \ldots, \alpha_n, \alpha \underset{\text{ARP}}{\vdash} \beta$$

en la que no se usen S_1 o I_0 respecto de variables de α, podemos construir una deducción

$$\gamma_1, \ldots, \gamma_r, \alpha_1, \ldots, \alpha_n \underset{\text{ARP}}{\vdash} \alpha \to \beta,$$

donde $\gamma_1, \ldots, \gamma_r$ son teoremas de ARP, *en la que sólo se usa S_1 o I_0 respecto de variables respecto a las que ya se usaban estas reglas en la deducción dada o bien respecto de variables que podemos elegir arbitrariamente.*

Naturalmente, los teoremas $\gamma_1, \ldots, \gamma_r$ se pueden eliminar del conjunto de premisas simplemente incluyendo sus demostraciones en la deducción, y así tenemos el enunciado del teorema que hemos dado. Sin embargo, la ventaja de esta formulación es que permite aplicar sucesivamente el teorema de deducción acumulando teoremas γ_i. En efecto:

Imaginemos que estamos construyendo una deducción $\delta_1, \ldots, \delta_m$ a partir de unas premisas $\alpha_1, \ldots, \alpha_n$ (que podemos suponer que están incluidas entre las fórmulas δ_i). En un momento dado añadimos una fórmula α como hipótesis y continuamos deduciendo sin usar S_1 o I_0 respecto de variables en α.

Luego añadimos α' como hipótesis y evitamos usar S_1 o I_0 respecto de variables en α' o en α, con lo que tenemos una deducción

$$\delta_1, \ldots, \delta_m, \alpha, \delta_{m+2}, \ldots, \delta_{m+m'}, \alpha', \delta_{m+m'+2} \ldots, \beta'.$$

En este punto aplicamos el teorema de deducción, que nos da una deducción

$$\gamma_1, \ldots, \gamma_r, \delta_1, \ldots, \delta_m, \alpha, \delta_{m+2}, \ldots, \delta_{m+m'} \underset{\text{ARP}}{\vdash} \alpha' \to \beta'$$

en la que podemos asegurar que no se usa S_1 o I_0 respecto de variables en α. Ahora prolongamos esta deducción —y en la práctica marcaremos las líneas comprendidas entre α' y β' con una línea vertical para indicar que ya no contamos con ellas— sin usar S_1 o I_0 respecto de variables en α, hasta llegar a

$$\gamma_1, \ldots, \gamma_r, \delta_1, \ldots, \delta_m, \alpha, \delta_{m+2}, \ldots, \beta.$$

Aquí podemos aplicar por segunda vez el teorema de deducción, que nos da una deducción

$$\gamma_1, \ldots, \gamma_{r+r'}, \delta_1, \ldots, \delta_m \underset{\text{ARP}}{\vdash} \alpha \to \beta.$$

Desde aquí —tras haber marcado las líneas entre α y β— continuamos deduciendo hasta llegar a

$$\gamma_1, \ldots, \gamma_{r+r'}, \delta_1, \ldots, \delta_m, \ldots, \epsilon$$

y con esto podemos concluir que

$$\alpha_1, \ldots, \alpha_n \underset{\text{ARP}}{\vdash} \epsilon,$$

sin más que incorporar a la deducción las demostraciones de los teoremas γ_i y las deducciones de las fórmulas δ_i a partir de las premisas.

Así hemos encadenado dos usos del teorema de deducción, pero igualmente podríamos haber encadenado cualquier número de usos. La clave está en que cada vez que lo aplicamos —en virtud del segundo enunciado— podemos asegurar que la deducción que obtenemos no usa las reglas S_1 o I_0 respecto de variables prohibidas por las aplicaciones que tenemos pendientes.

Más en general, ahora es claro que no debe preocuparnos insertar un teorema de ARP en una deducción en la que estemos aplicando el teorema de deducción por la posibilidad de que su demostración requiera usos prohibidos de S_1 o I_0, pues dicho teorema siempre puede incorporarse a la lista de teoremas γ_i y posponer la inclusión de su demostración hasta el momento en que ya hemos terminado de aplicar el teorema de deducción.

Lo que si tenemos que cuidar es no usar ninguna regla de inferencia derivada que requiera un uso de S_0 o I_0 respecto de variables prohibidas. Ciertamente, las reglas I_1, I_2 o I esconden un uso I_0 respecto de la misma variable, por lo que tampoco podemos usarlas respecto de variables prohibidas.

En cambio, no sucede lo mismo con las demás reglas de inferencia que hemos demostrado. En ninguna de ellas se usa I_0 y sólo se aplica S_1 a teoremas que podemos incluir en la lista de los γ_i respecto de variables que podemos elegir arbitrariamente.

Por ejemplo, para probar la primera regla del valor absoluto (página 43), partiendo de $s = t$, usamos S_2 para concluir que $|s - t| = |s - s|$ y, a partir del teorema $|x - x| = 0$ (donde la variable x la elegimos nosotros), por S_1 obtenemos que $|s - s| = 0$, y así concluimos $|s - t| = 0$.

Similarmente, para la segunda regla, si partimos de $|s - t| = 0$, concluimos $s = t$ aplicando S_1 al teorema $0 \dot{-} x = 0$ y a otras propiedades de la resta truncada, pero siempre respecto de variables x elegidas libremente (pues si un teorema vale para una variable, también es un teorema la fórmula que resulta de sustituirla por otra).

El lector puede comprobar fácilmente que todos los usos de S_1 en las pruebas de reglas de inferencia que hemos dado (salvo en las de inducción) son de este tipo y, por consiguiente, ninguna de ellas entra en conflicto con el uso del teorema de deducción. Más aún, en el caso de las reglas de inducción, sólo tenemos un uso de I_0 respecto de la misma variable y, a lo sumo, usos admisibles de S_1, es decir, aplicaciones de la regla a teoremas respecto de variables que podemos elegir. ∎

Ahora pasamos a demostrar el teorema y a continuación explicaremos por qué es necesaria la restricción sobre el uso de S_1 o I_0.

Demostración (del teorema de deducción): Sea $\delta_1, \ldots, \delta_m$ una deducción con premisas $\alpha_1, \ldots, \alpha_n, \alpha$ que termina en $\delta_m \equiv \beta$ y vamos a ver cómo construir la deducción requerida de $\alpha \to \beta$. Sea $R \equiv 1 \dot{-} t_\alpha$, que es un término en el que aparecen las mismas variables que en α.

Pongamos que $\delta_i \equiv s_i = t_i$, y vamos a demostrar que $\delta_i' \equiv R \cdot s_i = R \cdot t_i$ es

un teorema para todo i. En particular, para $i = m$ tendremos probado

$$|R \cdot s_m - R \cdot t_m| = 0,$$

de donde, por la propiedad 4 del valor absoluto, se sigue que $R \cdot |s_m - t_m| = 0$, que es lo mismo que $(1 \dot{-} t_\alpha) \cdot t_\beta = 0$, y esto es, por definición, $\alpha \to \beta$.

Además comprobaremos que en la deducción de δ'_i sólo se usan las reglas S_1 o I_0 respecto de variables respecto a las que ya se usaban estas reglas en la deducción de δ_i o bien respecto de variables que podemos elegir libremente y teoremas que podemos incorporar a la lista de los γ_i.

De hecho, en el paso de δ'_m a $\alpha \to \beta$ ya hemos necesitado algunas de estas aplicaciones: primero la primera regla del valor absoluto y luego tres aplicaciones de S_1 a la propiedad 4 del valor absoluto respecto de variables x, y, z que podemos elegir libremente.

Razonamos inductivamente, es decir, suponemos que tenemos ya deducciones de las fórmulas δ'_j, para $j < i$, y vamos a ver cómo construir una deducción de δ'_i en las condiciones requeridas. Para ello distinguimos varios casos:

Si $\delta_i \equiv \alpha$, entonces hay que probar

$$\delta'_i \equiv (1 \dot{-} |s_i - t_i|) \cdot s_i = (1 \dot{-} |s_i - t_i|) \cdot t_i,$$

y esta fórmula se deduce aplicando S_1 (respecto de variables x, y elegidas libremente) a la fórmula (1.5), que hemos demostrado en la prueba del teorema 1.19 (que incluimos entre los teoremas γ_i).

Si $\delta_i \equiv s_i = t_i$ es un axioma o una premisa α_j, entonces podemos incluir δ_i en nuestra deducción (como axioma o premisa) y $\delta'_i \equiv R \cdot s_i = R \cdot t_i$ se obtiene aplicando la regla S_2 al término $R \cdot x$.

Por consiguiente, basta probar que si δ_i se sigue de fórmulas anteriores de la deducción mediante una de las cuatro reglas de inferencia primitivas, entonces δ'_i puede deducirse de las fórmulas correspondientes de la demostración que estamos construyendo. Equivalentemente, basta probar que cada regla de inferencia sigue siendo válida si sus fórmulas se multiplican por R.

- Para la regla S_1 se trata de ver que

$$\frac{R \cdot s_1(x) = R \cdot s_2(x)}{R \cdot s_1(t) = R \cdot s_2(t)}$$

es una regla de inferencia válida, lo cual es cierto (es un caso particular de S_1) siempre y cuando la variable x no aparezca en R, y esto lo tenemos garantizado por la hipótesis del teorema, ya que las variables de R son las de la premisa α y suponemos que en la demostración nunca se usa S_1 respecto de variables presentes en α. Además, con esto estamos usando S_1 en la nueva deducción respecto de una variable respecto a la cual ya se usaba S_1 en la deducción original.

- Para la regla S_2 tenemos que demostrar que

$$\frac{R \cdot t_1 = R \cdot t_2}{R \cdot s(t_1) = R \cdot s(t_2)}$$

es una regla de inferencia válida. Vamos a probar, de hecho, el teorema

$$R \cdot t_1 = R \cdot t_2 \to R \cdot s(t_1) = R \cdot s(t_2). \tag{1.6}$$

La regla se sigue, entonces, de aplicar MP (y sólo tenemos que incorporar este teorema a la lista de los γ_i). En primer lugar observamos que

$$R \cdot t_1 = R \cdot t_2 \to (R = 0 \vee t_1 = t_2), \tag{1.7}$$

pues, por definición de la implicación (y la propiedad 4 del valor absoluto), se trata de la fórmula

$$(1 \dot{-} R \cdot |t_1 - t_2|) \cdot R \cdot |t_1 - t_2| = 0,$$

que es un caso particular de $(1 \dot{-} x) \cdot x = 0$.

Por otra parte,

$$R = 0 \to R \cdot s(t_1) = R \cdot s(t_2),$$

$$t_1 = t_2 \to R \cdot s(t_1) = R \cdot s(t_2).$$

La primera implicación se debe a que, por definición, es la fórmula

$$(1 \dot{-} R) \cdot R \cdot |s(t_1) - s(t_2)| = 0$$

y sabemos que $(1 \dot{-} R) \cdot R = 0$.

La segunda es un caso particular del teorema 1.19 aplicado al término $R \cdot z$, donde z es cualquier variable que no esté en R. Por la regla del dilema concluimos

$$(R = 0 \vee t_1 = t_2) \to R \cdot s(t_1) = R \cdot s(t_2),$$

y la regla del silogismo hipotético aplicada a (1.7) nos da la conclusión (1.6).

- Para la regla T tenemos que demostrar

$$\frac{R \cdot t_1 = R \cdot t_2 \qquad R \cdot t_1 = R \cdot t_3}{R \cdot t_2 = R \cdot t_3}$$

que no es sino un caso particular de T.

- Para la regla I_0 tenemos que probar:

$$\frac{R \cdot s_1(0) = R \cdot s_2(0),\ R \cdot s_1(Sx) = R \cdot h(x, s_1(x)),\ R \cdot s_2(Sx) = R \cdot h(x, s_2(x))}{R \cdot s_1(x) = R \cdot s_2(x)}$$

bajo el supuesto de que la variable x no está en R.

Veamos en primer lugar que la regla de inferencia siguiente es válida:

$$\frac{s_1 = t_1 \qquad s_2 = t_2}{s_1 = s_2 \to t_1 = t_2}$$

En efecto, por la regla S_2, de las premisas (aplicando S_1 a teoremas que incluimos en la lista de los γ_i respecto de variables elegidas libremente) obtenemos

$$|s_1 - s_2| = |t_1 - s_2|, \qquad |t_1 - s_2| = |t_1 - t_2|,$$

luego por transitividad concluimos que $|s_1 - s_2| = |t_1 - t_2|$.

De aquí, por S_2,

$$(1 \dot{-} |s_1 - s_2|) \cdot |t_1 - t_2| = (1 \dot{-} |s_1 - s_2|) \cdot |s_1 - s_2| = 0,$$

y esta última fórmula es $s_1 = s_2 \to t_1 = t_2$.

Por consiguiente, de las premisas de la regla de inducción deducimos

$$R \cdot h(x, s_1(x)) = R \cdot h(x, s_2(x)) \to R \cdot s_1(Sx) = R \cdot s_2(Sx).$$

Por otra parte, un caso particular de (1.6) es:

$$R \cdot s_1(x) = R \cdot s_2(x) \to R \cdot h(x, s_1(x)) = R \cdot h(x, s_2(x)),$$

y, encadenando las dos implicaciones, por la regla del silogismo hipotético obtenemos

$$R \cdot s_1(x) = R \cdot s_2(x) \to R \cdot s_1(Sx) = R \cdot s_2(Sx).$$

Como también contamos con la premisa $R \cdot s_1(0) = R \cdot s_2(0)$, la regla de inducción general I nos permite concluir $R \cdot s_1(x) = R \cdot s_2(x)$. Como ya hemos señalado, el uso de I conlleva únicamente un uso de I_0 respecto de la misma variable a la que se aplicaba I_0 a la deducción dada y aplicaciones típicas de S_1 a teoremas que podemos incorporar a la lista de los γ_i, luego no hemos incorporado ningún uso de I_0 respecto de ninguna variable "nueva". ∎

Usos incorrectos del teorema de deducción Pasamos ya a analizar por qué es esencial no usar las reglas S_1 o de inducción respecto de variables presentes en las hipótesis. En primer lugar mostramos un par de ejemplos de lo que puede pasar si no tenemos en cuenta estas restricciones. En el primero usamos indebidamente la regla S_1:

(1)	$x = 0$	Hipótesis
(2)	$1 = 0$	S_1, 1 (no permitida)
(3)	$x = 0 \to 1 = 0$	Falso teorema de deducción
(4)	$0 = 0 \to 1 = 0$	S_1, 3
(5)	$0 = 0$	Teorema
(6)	$1 = 0$	MP, 4, 5

Obtenemos así una "demostración" de que $1 = 0$, lo cual es imposible por el teorema de corrección, y esto demuestra que el teorema de deducción sería falso

sin la restricción sobre el uso de S_1. Veamos ahora un ejemplo en el que usamos indebidamente I_2:

(1)	$x = 1$	Hipótesis
(2)	$Sx = 2$	$S_2, 1$
(3)	$0 \cdot (2 \dot{-} 0) = 0$	Teorema
(4)	$Sx \cdot (2 \dot{-} Sx) = 2 \cdot (2 \dot{-} 2)$	$S_2, 2$
(5)	$2 \cdot (2 \dot{-} 2) = 0$	Teorema
(6)	$Sx \cdot (2 \dot{-} Sx) = 0$	$T, 4, 5$
(7)	$x \cdot (2 \dot{-} x) = 0$	$I_2, 3, 6$ (no permitida)
(8)	$x = 1 \to x \cdot (2 \dot{-} x) = 0$	Falso teorema de deducción
(9)	$1 = 1 \to 1 \cdot (2 \dot{-} 1) = 0$	$S_2, 8$
(10)	$1 = 1$	Teorema
(11)	$1 \cdot (2 \dot{-} 1) = 0$	MP 9, 10
(12)	$1 \cdot (2 \dot{-} 1) = 1$	Teorema
(12)	$1 = 0$	T 11, 12

Para entender (desde un punto de vista semántico) por qué fallan los razonamientos anteriores, comparemos con un uso correcto del teorema de deducción:

(1)	$x = 0$	Hipótesis
(2)	$y \cdot x = y \cdot 0$	S_2, 1
(3)	$y \cdot 0 = 0$	Definición de $\cdot$
(4)	$y \cdot x = 0$	T 2, 3
(7)	$x = 0 \to y \cdot x = 0$	

En cualquier fórmula de ARP, como $x = 0 \to y \cdot x = 0$, entendemos que las variables representan números naturales arbitrarios. Esta fórmula dice que, para cualquier par de números naturales x e y, es cierto que si el primero es 0, su producto es 0. El hecho de que la regla S_1 sea válida depende esencialmente de este hecho. Si es correcto sustituir una variable cualquiera por un término cualquiera en una fórmula cualquiera es porque lo que vale para un número x cualquiera vale también para un término t cualquiera.

Sin embargo, si queremos probar que $x = 0 \to y \cdot x = 0$ y suponemos $x = 0$, ahí ya no podemos entender que x representa un número natural cualquiera. Ahora x es un número concreto: el número 0. Si aplicamos S_1 para sustituir x por 1 llegamos al absurdo $1 = 0$ porque hemos sustituido una variable que representa a un número que cumple una propiedad específica (ser cero) por otro número que no cumple esa propiedad. Por eso falla el razonamiento.

Lo mismo sucede si intentamos aplicar inducción. Al haber supuesto $x = 1$, la variable x ya no representa un número arbitrario, sino un número que tiene una propiedad particular que no tienen todos los números. Sin embargo, para que la inducción

$$\frac{0 \cdot (2 \dot{-} 0) = 0 \qquad Sx \cdot (2 \dot{-} Sx) = 0}{x \cdot (2 \dot{-} x) = 0}$$

fuera válida, haría falta que la fórmula $Sx \cdot (2 \dot{-} Sx) = 0$ fuera cierta para todo número natural x, mientras que sólo la hemos demostrado bajo la hipótesis de

que $x = 1$, luego no es correcto concluir que $x \cdot (2 \dot{-} x) = 0$ vale para todo número natural.

En resumen, la prohibición de usar S_1 y las reglas de inducción respecto de variables de la hipótesis es la forma técnica de establecer que las variables de las hipótesis ya no representan números naturales arbitrarios, sino números particulares que cumplen la hipótesis, de modo que sustituir dichas variables por términos arbitrarios podría dar lugar a fórmulas falsas si dichos términos representan números que no cumplen la hipótesis, e igualmente aplicar la regla de inducción podría ser incorrecto porque no habríamos probado las premisas con la generalidad requerida. ∎

Del teorema de deducción se sigue inmediatamente que en ARP también podemos razonar por reducción al absurdo:

Teorema 1.21 (Reducción al absurdo) *Si se cumple*

$$\alpha_1, \ldots, \alpha_n, \neg\alpha \underset{\text{ARP}}{\vdash} \beta \wedge \neg\beta$$

y en la deducción no se usan las reglas S_1 o I_0 respecto de variables que aparecen en α, entonces

$$\alpha_1, \ldots, \alpha_n \underset{\text{ARP}}{\vdash} \alpha.$$

En otras palabras: para deducir una fórmula α podemos suponer $\neg\alpha$ y llegar a una contradicción $\beta \wedge \neg\beta$ (aunque en la práctica basta con llegar a las líneas β y $\neg\beta$ en la deducción, pues siempre se pueden combinar mediante IC).

Demostración: Por el teorema de deducción, de $\alpha_1 \ldots, \alpha_n$ se deduce la implicación $\neg\alpha \to (\beta \wedge \neg\beta)$, pero $\neg(\beta \wedge \neg\beta)$ es un teorema (es la propiedad 10 que hemos probado de los conectores), luego la regla del *modus tollens* nos da $\neg\neg\alpha$ y la propiedad 2 nos permite concluir α. ∎

Igualmente se justifica la variante consistente en suponer α y concluir $\neg\alpha$. Es claro que en la práctica podemos usar este teorema en las mismas condiciones que el teorema de deducción (intercalando incluso usos de uno y otro teorema).

Ahora ya estamos en condiciones de usar los conectores lógicos con naturalidad, sin depender de teoremas aritméticos sofisticados. Como ilustración terminamos probando algunos resultados elementales que requieren conectores para ser enunciados. El primero expresa que todo número natural no nulo es el siguiente de otro número natural:

Teorema 1.22 *Las fórmulas siguientes son teoremas de* ARP:

1. $x = 0 \vee x = S(x \dot{-} 1)$.
2. $x + z = y + z \to x = y$.
3. $x + y = 0 \leftrightarrow x = 0 \wedge y = 0$.
4. $xy = 0 \leftrightarrow x = 0 \vee y = 0$.

DEMOSTRACIÓN: 1) se prueba por inducción sobre x. Para $x = 0$ es trivial y, supuesto cierto para x, tenemos que $(Sx) \dot{-} 1 = (x + 1) \dot{-} 1 = x$, luego $Sx = S(Sx \dot{-} 1)$. He aquí una prueba detallada:

Demostración del teorema 1.22 1)

(1)	$0 = 0$	Teorema
(2)	$0 = 0 \vee 0 = S(0 \dot{-} 1)$	ID 1
(3)	$(x + y) \dot{-} x = y$	Teorema
(4)	$(x + 1) \dot{-} 1 = x$	S_1 3
(5)	$x + 1 = Sx$	Teorema
(6)	$(x + 1) \dot{-} 1 = Sx \dot{-} 1$	S_2 5
(7)	$x = Sx \dot{-} 1$	T 4, 6
(8)	$Sx = S(Sx \dot{-} 1)$	S_2 7
(9)	$Sx = 0 \vee Sx = S(Sx \dot{-} 1)$	ID 8
(10)	$x = 0 \vee x = S(x \dot{-} 1)$ $\to Sx = 0 \vee (Sx = S(Sx \dot{-} 1))$	$p \vdash q \to p$, 9
(11)	$x = 0 \vee x = S(x \dot{-} 1)$	I 2, 10

2) Se prueba fácilmente por inducción sobre z. Una prueba detallada es:

Demostración del teorema 1.22 2)

(1)	$x + 0 = x$	Definición de $+$
(2)	$y + 0 = y$	Definición de $+$
(3)	$x + 0 = y + 0$	Hipótesis
(4)	$x = y$	T 1, 2, 3
(5)	$x + 0 = y + 0 \to x = y$	
(6)	$x + z = y + z \to x = y$	Hipótesis
(7)	$x + Sz = y + Sz$	Hipótesis
(8)	$x + Sz = S(x + z)$	Definición de $+$
(9)	$y + Sz = S(y + z)$	Definición de $+$
(10)	$S(x + z) = S(y + z)$	T 7, 8, 9
(11)	$Su = Sv \to u = v$	Teorema
(12)	$S(x + z) = S(y + z) \to x + z = y + z$	S_1 11
(13)	$x = y$	Consecuencia lógica 10, 12, 6
(14)	$x + Sz = y + Sz \to x = y$	
(15)	$x + z = y + z \to x = y \to$ $x + Sz = y + Sz \to x = y$	
(16)	$x + z = y + z \to x = y$	I 5, 15

3) Es inmediato, porque si $y \neq 0$, entonces $y = S(y \dot{-} 1)$, luego

$$0 = x + y = S(x + (y \dot{-} 1)),$$

en contra del primer axioma de Peano, luego $y = 0$ y, por consiguiente, $x = 0$. La otra implicación es obvia. Una prueba detallada es:

Demostración del teorema 1.22 3)

(1)	$x + y = 0$	Hipótesis
(2)	$y \neq 0$	Hipótesis
(3)	$y = 0 \vee y = S(y \dot{-} 1)$	Teorema
(4)	$y = S(y \dot{-} 1)$	MTP 2, 3
(5)	$x + y = x + S(y \dot{-} 1)$	S_2, 4
(6)	$x + Sz = S(x + z)$	Definición de +
(7)	$x + S(y \dot{-} 1) = S(x + (y \dot{-} 1))$	S_1, 6
(8)	$S(x + (y \dot{-} 1)) = 0$	T 1, 5, 7
(9)	$Sz \neq 0$	Teorema
(10)	$S(x + (y \dot{-} 1)) \neq 0$	$S_1, 9$
(11)	$y = 0$	RA 8, 10
(12)	$x + y = x + 0$	S_1 11
(13)	$x + 0 = x$	Definición de +
(14)	$x = 0$	T1, 12, 13
(15)	$x = 0 \wedge y = 0$	IC 11, 14
(16)	$x + y = 0 \to x = 0 \wedge y = 0$	
(17)	$x = 0 \wedge y = 0$	Hipótesis
(18)	$x = 0$	EC 17
(19)	$y = 0$	EC 17
(20)	$x + y = x + 0$	S_2 19
(21)	$x + 0 = x$	Definición de +
(22)	$x + y = 0$	T 20, 21, 18
(23)	$x = 0 \wedge y = 0 \to x + y = 0$	
(24)	$x + y = 0 \leftrightarrow x = 0 \wedge y = 0$	IB 16, 23

4) Si $xy = 0$, pero $y \neq 0$, entonces $xy = x(S(y \dot{-} 1)) = x(y \dot{-} 1) + x = 0$, luego $x = 0$ por el resultado precedente. Dejamos como ejercicio al lector probar detalladamente este último teorema a partir del esbozo de prueba que hemos dado. A partir de este momento ya no daremos más pruebas detalladas a este nivel, sino que el lector debería ser capaz de desarrollar los argumentos o, mejor aún, de convencerse de que puede hacerse y que, precisamente por ello, no hace falta hacerlo. ∎

1.5 Cuantificadores acotados

Es frecuente definir el orden usual de los números naturales estableciendo que $x \leq y$ equivale a que existe un número natural z tal que $y = x + z$. Sin embargo, no podemos formalizar esta definición en ARP porque el lenguaje $\mathcal{L}_{\text{arp}}$ no tiene forma de expresar ese "existe un z". No obstante, esto no es completamente

cierto: en $\mathcal{L}_{\mathrm{arp}}$ podemos afirmar la existencia de un número natural cuando podemos calcularlo explícitamente, y esto es aplicable a nuestro caso: si $x \leq y$, el número z que cumple $x+z = y$ es precisamente $z = y \dot{-} x$, por lo que podemos dar la definición siguiente:

Definición 1.23 $x \leq y \equiv x + (y \dot{-} x) = y$.

Veamos que esta definición permite probar los hechos básicos sobre la ordenación de los números naturales (usamos sin referencias el teorema 1.22):

1. $x \leq x + y$.

   Por la propiedad 6 de la resta truncada.

2. $x \leq x$.

3. $x \leq y \wedge y \leq x \to x = y$.

   Tenemos que $x + (y \dot{-} x) = y$, $y + (x \dot{-} y) = x$, luego

   $$x + (y \dot{-} x) + (x \dot{-} y) = x,$$

   luego $(y \dot{-} x) + (x \dot{-} y) = 0$, luego $x \dot{-} y = 0 = y \dot{-} x$, luego

   $$x = x + (y \dot{-} x) = y.$$

4. $x \leq y \wedge y \leq z \to x \leq z$.

   Tenemos $x + (y \dot{-} x) = y$, $y + (z \dot{-} y) = z$, luego

   $$x + (y \dot{-} x) + (z \dot{-} y) = z,$$

   luego $x \leq z$ por (1).

5. $0 \leq x$.

6. $x \leq Sx$.

   Pues $x + (Sx \dot{-} x) = x + ((x + 1) \dot{-} x) = x + 1 = Sx$.

7. $x \leq y \vee y \leq x$.

   Lo probamos por inducción sobre y. Para $y = 0$ es claro. Si vale para y, en el caso $x \leq y$ tenemos $x \leq y \leq Sy$, mientras que si $y \leq x$, entonces $y + (x \dot{-} y) = x$. Si $x \dot{-} y = 0$, entonces $y = x$ y $x \leq Sy$. Si $x \dot{-} y \neq 0$, entonces $y + S(((x \dot{-} y) \dot{-} 1)) = x$, luego $Sy + ((x \dot{-} y) \dot{-} 1) = x$, luego $Sy \leq x$ por (1).

8. $y \leq Sx \leftrightarrow y \leq x \vee y = Sx$.

   Si $y \leq Sx$, entonces $y + (Sx \dot{-} y) = Sx$. Si $Sx \dot{-} y = 0$, queda $y = Sx$ y, en caso contrario, $y + S((Sx \dot{-} y) \dot{-} 1) = Sx$, luego $y + ((Sx \dot{-} y) \dot{-} 1) = x$, luego $y \leq x$, por (1).

   Si $y \leq x \vee y = Sx$, en el segundo caso es claro que $y \leq Sx$, mientras que en el primero $y + (x \dot{-} y) = x$, luego $y + S(x \dot{-} y) = Sx$, luego $y \leq Sx$, de nuevo por (1).

9. $x \leq y \leftrightarrow x + z \leq y + z$.

   Si $x \leq y$, entonces $x + (y \dot{-} x) = y$, luego $x + z + (y \dot{-} x) = y + z$, luego $x + z \leq y + z$ por (1).

   Si $x + z \leq y + z$, entonces $x \leq y \vee y \leq x$, pero si se da el segundo caso, por la parte ya probada, $y + z \leq x + z$, luego $x + z = y + z$, luego $x = y$, luego $x \leq y$.

10. $z \neq 0 \wedge xz = yz \to x = y$.

    Podemos suponer que $x \leq y$. Entonces $x + (y \dot{-} x) = y$, luego

    $$xz + (y \dot{-} x)z = yz = xz,$$

    luego $(y \dot{-} x)z = 0$, luego $y \dot{-} x = 0$, luego $x = y$.

11. $x \leq y \to xz \leq yz$.

    Pues $x + (y \dot{-} x) = y$, luego $xz + (y \dot{-} x)z = yz$, luego $xz \leq yz$.

12. $z \neq 0 \wedge xz \leq yz \to x \leq y$.

    Se cumple $x \leq y \vee y \leq x$, pero en el segundo caso $yz \leq xz$, luego $yz = xz$, luego $y = x$, luego $x \leq y$.

Definimos $x < y \equiv x \leq y \wedge x \neq y$. Las propiedades obvias de esta relación se deducen inmediatamente de las propiedades correspondientes de $\leq$ y de las del igualador.

Por ejemplo, ahora podemos precisar 6 y afirma que $x < Sx$, pues si fuera $x = Sx = x + 1$, sería $0 = 1 = S0$, en contradicción con el primer axioma de Peano.

De las propiedades de la relación de orden que hemos demostrado se siguen ya sin dificultad todas las propiedades elementales de uso habitual, como que $x \leq y \wedge x' \leq y' \to x + x' \leq y + y'$, etc.

Ahora estamos en condiciones de extender ligeramente la lógica de ARP:

Definición 1.24 Si $\alpha \equiv s_1 = s_2$ es una fórmula de $\mathcal{L}_{\mathrm{arp}}$ cuyas variables estén entre $x_1, \ldots, x_n$, definimos el funtor $\chi_\alpha(x_1, \ldots, x_n) \equiv \chi[\alpha]$ dado por

$$\chi_\alpha(x_1, \ldots, x_n) = 1 \dot{-} t_\alpha \equiv 1 \dot{-} |s_1 - s_2|.$$

Así, teniendo en cuenta las reglas (1.3), se cumple que

$$\chi_\alpha(x_1, \ldots, x_n) = 1 \leftrightarrow \alpha(x_1, \ldots, x_n),$$

$$\chi_\alpha(x_1, \ldots, x_n) = 0 \leftrightarrow \neg\alpha(x_1, \ldots, x_n).$$

Dada una fórmula $\alpha(x_1, \ldots, x_n, x)$, consideramos el funtor dado por las ecuaciones

$$M_\alpha(x_1, \ldots, x_n, 0) = \chi_\alpha(x_1, \ldots, x_n, 0),$$

$$M_\alpha(x_1,\dots,x_n,Sx) = M_\alpha(x_1,\dots,x_n,x)+$$
$$\chi[M_\alpha(x_1,\dots,x_n,x)=0 \wedge \chi_\alpha(x_1,\dots,x_n,Sx)=1]\cdot(x+2).$$

Notemos que —abreviando $x_1,\dots,x_n$ como $\bar{x}$— la sucesión

$$M_\alpha(\bar{x},0), \quad M_\alpha(\bar{x},1), \quad M_\alpha(\bar{x},2), \quad \dots$$

toma el valor 0 hasta que aparece el primer x que cumple $\alpha(\bar{x},x)$, y a partir de ese momento toma el valor $x+1$.

Definimos

$$\begin{aligned}
\bigvee u \le x\, \alpha(x_1,\dots,x_n,u) &\equiv M_\alpha(x_1,\dots,x_n,x)\neq 0,\\
\bigwedge u \le x\, \alpha(x_1,\dots,x_n,u) &\equiv \neg\bigvee u\le x\, \neg\alpha(x_1,\dots,x_n,u),\\
\mu u \le x\, \alpha(x_1,\dots,x_n,u) &\equiv M_\alpha(x_1,\dots,x_n,x) \dot{-} 1.
\end{aligned}$$
$$\overset{1}{\bigvee} u \le x\alpha(x_1,\dots,x_n,u) \equiv \bigvee u\le x\, \alpha(x_1,\dots,x_n,u)\wedge$$
$$\bigwedge u\le x \bigwedge v\le x(\alpha(x_1,\dots,x_n,u)\wedge\alpha(x_1,\dots,x_n,v)\to u=v).$$

Notemos que, a pesar de la notación que empleamos, ninguna de cuatro expresiones anteriores contiene la variable u.

Las consideraciones precedentes muestran que se cumple

$$\vDash \bigvee u\le x\, \alpha(x_1,\dots,x_n,u)[v]$$

si y sólo si existe un número natural $m\le v(x)$ tal que $\vDash \alpha[v^m_u]$, de modo que una transcripción al castellano de

$$\bigvee u\le x\, \alpha(x_1,\dots,x_n,u)$$

es

Existe un número natural $u\le x$ que cumple $\alpha(x_1,\dots,x_n,u)$.

Por consiguiente

$$\bigwedge u\le x\, \alpha(x_1,\dots,x_n,u)$$

es "No existe un $u\le x$ que no cumpla $\alpha(x_1,\dots,x_n,u)$.", que es lo mismo que

Para todo número natural $u\le x$ se cumple $\alpha(x_1,\dots,x_n,u)$.

Esto implica a su vez que $\overset{1}{\bigvee} u\le x\alpha(x_1,\dots,x_n,u)$ significa:

Existe un único $u\le x$ que cumple $\alpha(x_1,\dots,x_n,u)$.

Por último, es claro que $N(\mu u\le x\, \alpha(x_1,\dots,x_n,u))[v]$ es el mínimo número natural $m\le v(x)$ que cumple $\vDash\alpha[v^m_u]$, o bien 0 si no existe tal mínimo.

Pero esto es sólo el significado que tienen las fórmulas y el término que acabamos de definir. Ahora vamos a ver que podemos demostrar los hechos que cabe esperar. En los resultados siguientes escribiremos $\alpha(x)$ en lugar de $\alpha(x_1,\dots,x_n,x)$, de modo que no supondremos que x es la única variable de α.

1. $\mu u \le x\,\alpha(u) \le x$.

   Por inducción sobre x. Para $x = 0$ tenemos que

   $$\mu u \le 0\,\alpha(u) = M_\alpha(0) \dot{-} 1 = \chi_\alpha(0) \dot{-} 1 = 0 \le 0,$$

   pues $\chi_\alpha(0)$ sólo puede valer 0 o 1.

   Si vale para x, distinguimos dos casos:

   Si $M_\alpha(x) \neq 0$, entonces $M_\alpha(Sx) = M_\alpha(x)$, luego

   $$\mu u \le Sx\,\alpha(u) = \mu u \le x\,\alpha(u) \le x \le Sx.$$

   Si $M_\alpha(x) = 0$, entonces

   $$M_\alpha(Sx) = \chi[M_\alpha(x) = 0 \wedge \chi_\alpha(Sx) = 1] \cdot (x + 2) \le x + 2,$$

   luego $\mu u \le Sx\,\alpha(u) = M_\alpha(Sx) \dot{-} 1 \le (x + 2) \dot{-} 1 = Sx$.

2. $y \le x \wedge \alpha(y) \to (\bigvee u \le x\,\alpha(u) \wedge \mu u \le x\,\alpha(u) \le y)$.

   Por inducción sobre x. Para $x = 0$ tenemos $y = 0$, $\alpha(0)$, luego

   $$M_\alpha(0) = \chi_\alpha(0) = 1 \neq 0,$$

   luego $\bigvee u \le 0\,\alpha(u)$. Además, $\mu u \le 0\,\alpha(u) = M_\alpha(0) \dot{-} 1 = 1 \dot{-} 1 = 0$, luego $\mu u \le 0\,\alpha(u) \le y$.

   Supuesto para x, si $y \le Sx \wedge \alpha(y)$, o bien $y \le x$ o bien $y = Sx$. En el primer caso, por hipótesis de inducción, $\bigvee u \le x\,\alpha(u)$, es decir, $M_\alpha(x) \neq 0$, luego $M_\alpha(Sx) \neq 0$, luego también se cumple $\bigvee u \le Sx\,\alpha(u)$.

   Además, $M_\alpha(Sx) = M_\alpha(x)$, luego $\mu u \le Sx\,\alpha(u) = \mu u \le x\,\alpha(u) \le y$.

   Si $y = Sx$, tenemos $\chi_\alpha(Sx) = 1 \neq 0$ luego, o bien $M_\alpha(x) \neq 0$, o bien

   $$\chi[M_\alpha(x) = 0 \wedge \chi_\alpha(Sx) = 1] \cdot (x + 2) \neq 0,$$

   y en ambos casos $M_\alpha(Sx) \neq 0$, luego $\bigvee u \le Sx\,\alpha(u)$.

   Si $M_\alpha(x) \neq 0$, entonces $M_\alpha(Sx) = M_\alpha(x)$, luego

   $$\mu u \le Sx\,\alpha(u) = \mu u \le x\,\alpha(u) \le x \le Sx.$$

   Si $M_\alpha(x) = 0$, entonces

   $$M_\alpha(Sx) = \chi[M_\alpha(x) = 0 \wedge \chi_\alpha(Sx) = 1] \cdot (x + 2) = x + 2,$$

   luego $\mu u \le Sx\,\alpha(u) = Sx \le y$.

3. $\bigvee u \le x\,\alpha(u) \to \alpha(\mu u \le x\,\alpha(u))$.

   Por inducción sobre x. Para $x = 0$ tenemos $\chi_\alpha(0) = M_\alpha(0) \neq 0$, luego se cumple $\alpha(0)$ y además $\mu u \le 0\,\alpha(u) = M_\alpha(0) \dot{-} 1 = 1 \dot{-} 1 = 0$, luego se cumple $\alpha(\mu u \le 0\,\alpha(u))$.

Si vale para x y $\bigvee u \le Sx\,\alpha(u)$, entonces $M_\alpha(Sx) \neq 0$. Distinguimos dos casos:

Si $M_\alpha(x) \neq 0$, entonces $\bigvee u \le x\,\alpha(u)$, luego por hipótesis de inducción $\alpha(\mu u \le x\,\alpha(u))$. Además $\mu u \le x\,\alpha(u) = M_\alpha(x) \dot{-} 1$. En este caso $M_\alpha(Sx) = M_\alpha(x)$, luego $\mu u \le Sx\,\alpha(u) = \mu u \le x\,\alpha(u)$, luego $\alpha(\mu u \le Sx\,\alpha(u))$.

Si $M_\alpha(x) = 0$, necesariamente

$$\chi[M_\alpha(x) = 0 \wedge \chi_\alpha(Sx) = 1] \cdot (x+2) \neq 0,$$

luego $\chi_\alpha(Sx) = 1$, luego se cumple $\alpha(Sx)$. Además

$$M_\alpha(Sx) = \chi[M_\alpha(x) = 0 \wedge \chi_\alpha(Sx) = 1] \cdot (x+2) = x+2,$$

luego $\mu u \le Sx\,\alpha(u) = (x+2) \dot{-} 1 = Sx$, luego $\alpha(\mu u \le Sx\,\alpha(u))$.

Recapitulando lo que hemos obtenido, por una parte podemos introducir el particularizador y el mínimo así:

$$y \le x \wedge \alpha(y) \to (\bigvee u \le x\,\alpha(u) \wedge \mu u \le x\alpha(u) \le y \wedge \alpha(\mu u \le x\,\alpha(u)))$$

y, por otra, podemos eliminar el particularizador así:

$$\bigvee u \le x\,\alpha(u) \to (\mu u \le x\alpha(u) \le x \wedge \alpha(\mu u \le x\,\alpha(u))).$$

Para el cuantificador universal tenemos, por una parte, la eliminación:

4. $y \le x \wedge \bigwedge u \le x\,\alpha(u) \to \alpha(y)$.
   En efecto, si $y \le x \wedge \bigwedge u \le x\,\alpha(u)$ y $\neg\alpha(y)$, entonces $\bigvee u \le x\,\neg\alpha(u)$, luego $\neg\bigwedge u \le x\,\alpha(u)$ y tenemos una contradicción.

En cambio, la introducción del generalizador tiene que formularse como regla de inferencia:

$$\text{(IG)}\ \frac{y \le x \to \alpha(y)}{\bigwedge u \le x\,\alpha(u)}$$

En efecto, si suponemos la premisa, pero negamos la conclusión, tenemos que $\bigvee u \le x\,\neg\alpha(u)$, luego $\mu u \le x\,\neg\alpha(u) \le x$ y $\neg\alpha(\mu u \le x\,\neg\alpha(u))$, pero la premisa nos da entonces que $\alpha(\mu u \le x\,\neg\alpha(u))$ y tenemos una contradicción. ∎

Nota Observemos que en la prueba de la regla IG aplicamos a la premisa la regla S_1 respecto de la variable y, lo cual hace que no podamos aplicar el teorema de deducción para concluir que la implicación

$$(y \le x \to \alpha(y)) \to \bigwedge u \le x\,\alpha(u)$$

sea un teorema de ARP. De hecho, es fácil deducir una contradicción de este presunto teorema.

Por el mismo motivo, no es lícito usar IG en un entorno en el que esté prohibido usar S_1 respecto de la variable y. ∎

Podemos considerar también las variantes

$$\bigwedge u < x\, \alpha(u) \equiv \bigwedge u \leq x(u < x \to \alpha(u)),$$

$$\bigvee u < x\, \alpha(u) \equiv \bigvee u \leq x(u < x \wedge \alpha(u)),$$

y a partir de los resultados ya probados es fácil probar que cumplen los resultados obvios que cabe esperar que cumplan.

Inducción completa Ahora es fácil probar la regla de inducción completa:

$$\frac{\bigwedge u < x\, \phi(u) \to \phi(x)}{\phi(x)}$$

DEMOSTRACIÓN: Probamos por inducción que $\psi(x) \equiv \bigwedge u < x\, \phi(u)$. Se cumple trivialmente para $x = 0$ y, si vale para x, por la premisa, tenemos $\phi(x)$, de donde se deduce claramente que $\bigwedge u < Sx\, \phi(u)$. En particular se cumple $\psi(Sx) \equiv \bigwedge u < Sx\, \phi(u)$, luego haciendo $u = x$ obtenemos $\phi(x)$. ∎

Nota Como en la prueba de la regla de inducción completa se usa la regla de inducción sobre la variable x, no podemos usar esta regla en contextos en los que no podemos usar la regla I_0 sobre x. ∎

Con esto tenemos ya completamente expuesta la lógica de la Aritmética Recursiva Primitiva. En el capítulo siguiente veremos cómo podemos trabajar en ella.

Capítulo II

Elementos de aritmética

En el capítulo anterior hemos definido la Aritmética Recursiva Primitiva, una teoría axiomática con nombres para todos los números naturales (numerales) y todas las funciones recursivas primitivas (funtores). Recordemos los principales teoremas aritméticos que hemos demostrado en ella. En primer lugar tenemos los axiomas de Peano:

1. $Sx \neq 0$,
2. $Sx = Sy \to x = y$,
3. $x + 0 = x$,
4. $x + Sy = S(x + y)$,
5. $x \cdot 0 = 0$,
6. $x \cdot Sy = x \cdot y + x$.

De la definición de suma se deduce en particular, llamando $1 \equiv S0$, que $Sx = x + 1$, y a partir de ahora ya no usaremos nunca de forma explícita el funtor S, sino que escribiremos $x + 1$ en lugar de Sx. Así, el principio de inducción, que se considera parte de los axiomas de Peano, se enuncia en ARP mediante la regla de inferencia

$$(I)\ \frac{\alpha(0) \qquad \alpha(x) \to \alpha(x+1)}{\alpha(x)}.$$

También tenemos a nuestra disposición las reglas de inferencia S_1, S_2, T, que permiten probar cualquier cosa razonable relacionada con el igualador, así como las reglas de inferencia que regulan los conectores lógicos.

Entre las propiedades de la suma y el producto tenemos:

1. $(x + y) + z = x + (y + z)$,
2. $x + y = y + x$,

3. $x + z = y + z \to x = y$.
4. $x + y = 0 \leftrightarrow x = 0 \wedge y = 0$.
5. $x \cdot 1 = x$,
6. $x \cdot y = y \cdot x$,
7. $x \cdot (y + z) = x \cdot y + x \cdot z$,
8. $x(yz) = (xy)z$,
9. $xy = 0 \leftrightarrow x = 0 \vee y = 0$,
10. $z \neq 0 \wedge xz = yz \to x = y$.

A partir de estas propiedades se pueden demostrar fácilmente (es decir, con los argumentos que emplearía habitualmente un matemático, sin ningún artificio lógico necesario por estar trabajando en ARP) todos los hechos elementales que nos permiten manipular expresiones aritméticas que involucran sumas y productos. Para la relación de orden tenemos:

1. $x \leq x$,
2. $x \leq y \wedge y \leq x \to x = y$,
3. $x \leq y \wedge y \leq z \to x \leq z$,
4. $x \leq y \vee y \leq x$,
5. $0 \leq x$,
6. $x \leq x + y$,
7. $y \leq x + 1 \leftrightarrow y \leq x \vee y = x + 1$,
8. $x \leq y \leftrightarrow x + z \leq y + z$,
9. $x \leq y \to xz \leq yz$,
10. $z \neq 0 \wedge xz \leq yz \to x \leq y$.

Hemos definido también $x < y \equiv x \leq y \wedge x \neq y$ y las propiedades de esta relación estricta se deducen fácilmente de las anteriores.

Nuevamente, a partir de estas propiedades se pueden demostrar de forma natural todas las propiedades básicas sobre manipulación de expresiones aritméticas que involucren sumas, productos y la relación de orden.

También tenemos definida la resta truncada, determinada por los dos teoremas siguientes:

1. $x \leq y \to x + (y \dot{-} x) = y$.
2. $y \leq x \to y \dot{-} x = 0$.

En efecto, el primero se sigue de la definición que hemos dado de $x \leq y$, y el segundo se debe a que si $y \leq x$, entonces $y + (x \dot{-} y) = x$, luego tenemos que $y \dot{-} x = y \dot{-} (y + (x \dot{-} y)) = 0$ por la propiedad 8 de la resta truncada.

Notemos que

$$x \leq y \wedge y \leq z \to z \dot{-} y \leq z \dot{-} x.$$

En efecto, si fuera $z \dot{-} x < z \dot{-} y$, tendríamos que

$$y + z = x + y + (z \dot{-} x) < x + y + (z \dot{-} y) = x + z,$$

luego $y < x$.

Tenemos definida la función característica de una fórmula α, determinada por los teoremas

$$\chi_\alpha(x_1, \ldots, x_n) = 1 \leftrightarrow \alpha(x_1, \ldots, x_n),$$

$$\chi_\alpha(x_1, \ldots, x_n) = 0 \leftrightarrow \neg\alpha(x_1, \ldots, x_n).$$

Con ella podemos definir funtores por casos. Por ejemplo, podemos definir

$$f(x_1, \ldots, x_n) = \begin{cases} g(x_1, \ldots, x_n) & \text{si } \alpha(x_1, \ldots, x_n), \\ h(x_1, \ldots, x_n) & \text{si } \neg\alpha(x_1, \ldots, x_n). \end{cases}$$

Esto ha de entenderse como

$$f(x_1, \ldots, x_n) = g(x_1, \ldots, x_n)\chi_\alpha(x_1, \ldots, x_n) + h(x_1, \ldots, x_n)\chi_{\neg\alpha}(x_1, \ldots, x_n)$$

y es fácil probar entonces los teoremas

$$\alpha(x_1, \ldots, x_n) \to f(x_1, \ldots, x_n) = g(x_1, \ldots, x_n),$$

$$\neg\alpha(x_1, \ldots, x_n) \to f(x_1, \ldots, x_n) = h(x_1, \ldots, x_n).$$

Por último, tenemos definidos los cuantificadores acotados

$$\bigvee u \leq x\, \alpha(x_1, \ldots, x_n, u), \quad \bigwedge u \leq x\, \alpha(x_1, \ldots, x_n, u), \quad \overset{1}{\bigvee} u \leq x\, \alpha(x_1, \ldots, x_n, u),$$

y el mínimo acotado $\mu u \leq x\, \alpha(x_1, \ldots, x_n, u)$, que cumplen todas las propiedades que cabe esperar que cumplan.

2.1 Más aritmética en ARP

Potencias Para completar la aritmética básica de los números naturales introducimos la exponenciación, que ya habíamos mostrado como ejemplo en el capítulo anterior, pero no habíamos probado nada sobre ella. Recordemos que la definición es:

$$x^0 = 1, \qquad x^{y+1} = x^y \cdot x.$$

Las propiedades siguientes se prueban fácilmente por inducción:

1. $x^{y+z} = x^y x^z$,
2. $(xy)^z = x^z y^z$,
3. $(x^y)^z = x^{yz}$,
4. $x > 0 \to 0^x = 0$,
5. $1^x = x$,
6. $x \le y \to x^z \le y^z$,
7. $x \ge 1 \wedge y \le z \to x^y \le x^z$,
8. $x \ge 2 \to y < x^y$.

Veamos la última como ejemplo. Razonamos por inducción sobre y. Para $y = 0$ es inmediato. Supuesto cierto para y, o bien $y = 0$, en cuyo caso se cumple que $y + 1 = 1 < x = x^1 = x^{y+1}$, o bien $y \ge 1$, en cuyo caso

$$x^{y+1} = x^y \cdot x \ge x^y \cdot 2 > y \cdot 2 = y + y \ge y + 1.$$

División euclídea El lector sabrá sin duda que, dados un dividendo D y un divisor $d \neq 0$, existen un único cociente c y un único resto r de modo que $D = dc + r$, con $0 \le r < d$. En el lenguaje de ARP no podemos decir "existen", pero tenemos dos alternativas:

Una es observar que el cociente y el resto tienen que ser menores o iguales que el dividendo, por lo que podemos decir $\bigvee c, r \le D$. La otra alternativa es definir funtores concretos que calculen el cociente y el resto, y eso es lo que vamos a hacer:

Definición 2.1 Definimos los funtores determinados por

$$c(D,d) = \mu u \le D\, D < d(u+1), \qquad r(D,d) = D \dot{-} d \cdot c(D,d).$$

Teorema 2.2 *Se cumple:*

1. $d \neq 0 \to D = d \cdot c(D,d) + r(D,d) \wedge r(D,d) < d$.
2. $d \neq 0 \wedge D = d \cdot c + r \wedge r < d \to c = c(D,d) \wedge r = r(D,d)$.

Demostración: 1) Como $d \neq 0$, tenemos que $1 \le d$, luego

$$D \le dD < dD + d = d(D+1),$$

y esto prueba que $\bigvee u \le D\, D < d(u+1)$, luego $D < d \cdot (c(D,d)+1)$. Tiene que ser $d \cdot c(D,d) \le D$, pues si fuera $D < d \cdot c(D,d)$, entonces $c(D,d) \neq 0$, luego, llamando $c' \equiv c(D,d) \dot{-} 1$, se cumpliría $c(D,d) = c' + 1$, luego $c' < c(D,d)$ y $D < d \cdot (c'+1)$, en contra de la minimalidad de $c(D,d)$. Así pues:

$$d \cdot c(D,d) \le D < d \cdot (c(D,d)+1).$$

Por la definición de $\leq$, la primera desigualdad implica que

$$D = d \cdot c(D,d) + r(D,d).$$

Tiene que ser $r(D,d) < d$, pues si $d \leq r(D,d)$, entonces

$$d + (r(D,d) \dot{-} d) = r(D,d)$$

y así

$$D = d \cdot c(D,d) + d + (r(D,d) \dot{-} d) = d \cdot (c(D,d)+1) + (r(D,d) \dot{-} d),$$

luego $d \cdot (c(D,d)+1) \leq D$, contradicción.

2) Tenemos que $dc \leq D = dc+r < dc+d = d(c+1)$, luego la minimalidad de $c(D,d)$ implica que $c(D,d) \leq c$. Si la desigualdad fuera estricta, $c(D,d)+1 \leq c$, luego

$$D < d(c(D,d)+1) \leq dc \leq dc + r = D,$$

y tenemos una contradicción.

Por lo tanto, $c = c(D,d)$, y como

$$d \cdot c(D,d) + r(D,d) = D = d \cdot c + r = d \cdot c(D,d) + r,$$

también $r = r(D,d)$. ∎

Por ejemplo, ahora podemos clasificar los números naturales en *pares* e *impares,* según si $r(D,2) = 0$ o $r(D,2) = 1$. Equivalentemente, por la unicidad de la división euclídea, los números pares son los de la forma $2n$ y los impares los de la forma $2n+1$. Es fácil probar que

$$r(ab,2) = r(a,2)r(b,2),$$

de modo que el producto de par por par o par por impar es par, mientras que el producto de impares es impar.

Pares ordenados Es posible numerar todos los pares de números naturales, es decir, asignar un número natural $\langle x,y\rangle_2$ a cada par de números naturales x, y (en un cierto orden) de modo que pares distintos tengan asignados números distintos y cada número natural tenga asignado un par. Una forma de hacerlo lo ilustra el diagrama siguiente:

$\vdots$						
4	10					
3	6	11				
2	3	7	12			
1	1	4	8	13		
0	0	2	5	9	14	
	0	1	2	3	4	$\cdots$

El número $\langle x, y\rangle_2$ es el situado en la columna x y en la fila y del rectángulo. Por ejemplo, vemos que $\langle 2, 1\rangle_2 = 8$.

La diagonal que contiene, por ejemplo, al $13 = \langle 3, 1\rangle_2$ contiene todos los pares cuyas componentes suman $x + y = 4$. Para llegar a ella hay que pasar antes por las diagonales anteriores, que contienen $1 + 2 + 3 + 4 = 10$ números, pero como empezamos en el 0 resulta que el $10 = \langle 0, 4\rangle$ es ya el primero de dicha diagonal. Para llegar a $\langle 3, 1\rangle_2$ hemos de avanzar $x = 3$ posiciones. En general, el par $\langle x, y\rangle_2$ se alcanza en la posición

$$z = 1 + 2 + \cdots + (x + y) + x = \frac{(x + y)(x + y + 1)}{2} + x.$$

Esto nos lleva a la definición siguiente:

Definición 2.3 Definimos el funtor

$$\langle x, y\rangle_2 = c((x + y)(x + y + 1), 2) + x.$$

Es fácil ver que la división es exacta,[1] de modo que

$$2\,\langle x, y\rangle_2 = (x + y)(x + y + 1) + 2x.$$

Observemos ahora que

$$(z + 1)(z + 2) = z^2 + 3z + 2 \geq 2z + 2 > 2z.$$

Por lo tanto, si definimos el funtor

$$F(z) = \mu u \leq z + 1\ (2z < u(u + 1)),$$

tenemos que $z + 1$ cumple la propiedad considerada, luego

$$2z < F(z)(F(z) + 1) \qquad \text{y} \qquad F(z) \leq z + 1.$$

No puede ser $F(z) = 0$, luego si definimos $G(z) = F(z) \dot{-} 1$, se cumple que $F(z) = G(z)+1$ y, como $G(z) < F(z)$, no puede cumplir la propiedad que define a F, luego

$$G(z)(G(z) + 1) \leq 2z < (G(z) + 1)(G(z) + 2), \qquad G(z) \leq z.$$

Observemos además que si un r cumple

$$r(r + 1) \leq 2z < (r + 1)(r + 2),$$

necesariamente $r = G(z)$, pues por la minimalidad de F tenemos $F(z) \leq r + 1$, luego $G(z) \leq r$ y, si la desigualdad fuera estricta, $G(z) + 1 \leq r$, luego

$$(G(z) + 1)(G(z) + 2) \leq r(r + 1) \leq 2z,$$

y tenemos una contradicción.

[1] Porque el dividendo es necesariamente el producto de un número par por un impar, luego es par.

Definimos el funtor

$$z_1 = c(2z \dot{-} G(z)(G(z)+1), 2).$$

Es fácil ver que el argumento de c es par, por lo que

$$2z = G(z)(G(z)+1) + 2z_1.$$

Además se cumple que $z_1 \leq G(z)$, pues si fuera $G(z) < z_1$, entonces

$$2G(z) < 2z_1 = 2z \dot{-} G(z)(G(z)+1),$$

luego $2G(z) + G(z)(G(z)+1) < 2z$, luego $G(z)^2 + 3G(z) + 1 \leq 2z$. Otra vez, distinguiendo casos según si $G(z)$ es par o impar, se comprueba que el miembro izquierdo siempre es impar, por lo que no puede darse la igualdad, luego

$$(G(z)+1)(G(z)+2) = G(z)^2 + 3G(z) + 2 \leq 2z,$$

en contradicción con la construcción de G.

Por lo tanto, $z_1 \leq G(z)$, luego el funtor

$$z_2 = G(z) \dot{-} z_1$$

cumple $G(z) = z_1 + z_2$ y

$$2z = G(z)(G(z)+1) + 2z_1 = (z_1+z_2)(z_1+z_2+1) + 2z_1 = 2\langle z_1, z_2\rangle_2,$$

luego $z = \langle z_1, z_2\rangle_2$.

El teorema siguiente recoge las propiedades que hemos demostrado de los funtores $\langle x, y\rangle_2$, z_1 y z_2 junto con algunas más:

Teorema 2.4 *Se cumple:*

1. $z = \langle z_1, z_2\rangle_2$,
2. $z_1 + z_2 \leq z$,
3. $z = \langle x, y\rangle_2 \to x = z_1 \wedge y = z_2$.

Demostración: Ya hemos demostrado la primera afirmación, y también la segunda, pues hemos visto que $z_1 + z_2 = G(z) \leq z$.

Si $z = \langle x, y\rangle_2$, llamando $r = x + y$, tenemos que $2z = r(r+1) + 2x$, con $x \leq r$, luego

$$r(r+1) \leq 2z \leq r^2 + r + 2x \leq r^2 + 3r < r^2 + 3r + 2 = (r+1)(r+2),$$

y antes hemos probado que esto implica $r = G(z) = z_1 + z_2$, luego

$$(z_1+z_2)(z_1+z_2+1) + 2z_1 = 2z = r(r+1) + 2x$$

implica que $2x = 2z_1$, luego $x = z_1$, y entonces $z_1 + z_2 = x + y$ implica que $y = z_2$. ∎

Con esto hemos probado que cada número natural z se expresa de forma única como un par $\langle z_1, z_2\rangle_2$ de números naturales. Tanto la función que a cada par de números naturales le asigna el número $\langle z_1, z_2\rangle_2$ como las proyecciones que a cada z le asignan sus coordenadas z_1 y z_2 son funciones recursivas primitivas.

Sucesiones finitas Similarmente podemos considerar el funtor

$$\langle x, y, z\rangle_3 = \langle \langle x, y\rangle_2, z\rangle_2$$

y las proyecciones $z_1^3 = (z_1)_1$, $z_2^3 = (z_1)_2$, $z_3^3 = z_2$, de modo que

$$z = \langle z_1, z_2, z_3\rangle, \qquad z = \langle u, v, w\rangle_3 \to u = z_1 \wedge v = z_2 \wedge w = z_3.$$

Así, podemos identificar los números naturales con las ternas de números naturales. En general, podemos definir los funtores

$$\langle x_1, \ldots, x_n\rangle_n = \big\langle \langle x_1, \ldots, x_{n-1}\rangle_{n-1}, x_n\big\rangle_2$$

y las proyecciones[2]

$$\begin{aligned} p_i^n(z) &= p_i^{n-1}(z_1), \qquad \text{para } i < n \\ p_n^n(z) &= z_2. \end{aligned}$$

Si definimos $\langle x\rangle_1 = x$, estas definiciones coinciden con las que ya teníamos en el caso $n = 2$.

Razonando por inducción sobre n, se prueba sin dificultad que

$$z = \langle p_1^n(z), \ldots, p_n^n(z)\rangle_n, \quad z = \langle x_1, \ldots, x_n\rangle_n \to x_1 = z_1 \wedge \cdots \wedge x_n = z_n.$$

Ejemplo Un simple cálculo nos da que

$$\langle 3, 1\rangle_2 = 13, \quad \langle 3, 1, 5\rangle_3 = \langle 13, 5\rangle_2 = 184,$$

de modo que el número 184 puede verse como él mismo, o como el par $\langle 13, 5\rangle_2$ o como la terna $\langle 3, 1, 5\rangle_3$.

En general, cada número natural n puede verse indistintamente como un número (él mismo), como un par de números, o una terna, o una cuádrupla, etc. Por ejemplo, el número $n = 17\,337\,210$ puede verse como

$$17\,337\,210, \quad (5\,882, 5), \quad (104, 3, 5), \quad (13, 0, 3, 5), \quad (3, 1, 0, 3, 5),$$

$$(0, 2, 1, 0, 3, 5), \quad (0, 0, 2, 1, 0, 3, 5), \quad (0, 0, 0, 2, 1, 0, 3, 5), \ldots$$

Para evitar que un mismo número pueda verse como muchas cosas a la vez, podemos definir una nueva familia de infinitos funtores:

$$\langle x_1, \ldots, x_n\rangle_\infty^n = \langle n \dot{-} 1, \langle x_1, \ldots, x_n\rangle_n\rangle_2 + 1.$$

Así, cada número natural no nulo s codifica una única sucesión. Para calcularla, pasamos a $s \dot{-} 1$, interpretamos el resultado como un par y la primera componente $+1$ nos indica la longitud n de la sucesión codificada, y ésta es la segunda componente interpretada precisamente como sucesión de longitud n.

[2]Notemos que no hemos definido un funtor $\langle x_1, \ldots, x_n\rangle_n$ de rango $n + 1$, sino infinitos funtores de rango n. Igualmente, la definición de las proyecciones no es la definición de un único funtor $p_i^n(z)$ de rango 3, sino que estamos definiendo infinitos funtores de rango 1, cada uno con su propia definición.

Podemos considerar que el número natural 0 codifica la sucesión vacía que tiene 0 términos.

Definición 2.5 Definimos la *longitud* de un número natural como

$$\ell(s) = (1 \dot{-} (1 \dot{-} s))((s \dot{-} 1)_1 + 1).$$

Así, $\ell(0) = 0$, mientras que si $s \neq 0$, entonces $\ell(s) > 0$ y $(s \dot{-} 1)_1 = \ell(s) \dot{-} 1$.

Para definir las proyecciones definimos primero el funtor

$$R(s, 0) = (s \dot{-} 1)_2, \qquad R(s, i + 1) = R(s, i)_1.$$

Así podemos definir:

$$p_i^\infty(s) = (1 \dot{-} i)R(s, (s \dot{-} 1)_1) + (1 \dot{-} (1 \dot{-} i))R(s, (s \dot{-} 1)_1 \dot{-} i)_2.$$

Notemos que $p_i^\infty(s)$ es un funtor de rango 2, es decir, que ahora no tenemos infinitos funtores de rango 1, sino que tanto i como s son argumentos de un mismo funtor.

Ejemplo Si $s = \langle 3, 2, 7\rangle_\infty = \left\langle 2, \langle\langle 3, 2\rangle_2, 7\rangle_2\right\rangle_2 + 1$, tenemos que

$$R(s, 0) = \langle\langle 3, 2\rangle_2, 7\rangle_2, \quad R(s, 1) = \langle 3, 2\rangle_2, \quad R(s, 2) = 3,$$

luego $p_0^\infty(s) = R(s, 2) = 3$, $p_1^\infty(s) = R(s, 1)_2 = 2$, $p_2^\infty(s) = R(s, 0)_2 = 7$. ∎

Nota A partir de ahora escribiremos $s_i \equiv p_i^\infty(s)$. ∎

Veamos que todo número natural, visto como sucesión, está unívocamente determinado por sus proyecciones.

Teorema 2.6 $\ell(s) = \ell(t) \wedge (\bigwedge i < \ell(s)\, s_i = t_i) \to s = t$.

Demostración: Si $\ell(s) = \ell(t) = 0$, entonces $s = t = 0$. Supongamos que $\ell(s) = \ell(t) > 0$. Entonces

$$l = (s \dot{-} 1)_1 = \ell(s) \dot{-} 1 = \ell(t) \dot{-} 1 = (t \dot{-} 1)_1.$$

Veamos por inducción que

$$i \leq l \to R(s, l \dot{-} i) = R(t, l \dot{-} i).$$

Para $i = 0$ tenemos que

$$R(s, l) = s_0 = t_0 = R(t, l).$$

Si es cierto para $i < l$, entonces $i + 1 \leq l$, luego $i + 1 + (l \dot{-} (i + 1)) = l$, de donde $(l \dot{-} (i + 1)) + 1 = l \dot{-} i$. Por lo tanto,

$$R(s, l \dot{-} (i{+}1))_1 = R(s, (l \dot{-} (i{+}1)){+}1) = R(s, l \dot{-} i) = R(t, l \dot{-} i) = R(t, l \dot{-} (i{+}1))_1.$$

Por otra parte,

$$R(s, l \dot{-} (i+1))_2 = s_{i+1} = t_{i+1} = R(t, l \dot{-} (i+1))_2,$$

luego $R(s, l \dot{-} (i+1)) = R(t, l \dot{-} (i+1))$. Esto completa la inducción y, aplicándolo a $i = l$ obtenemos

$$(s \dot{-} 1)_2 = R(s, 0) = R(t, 0) = (t \dot{-} 1)_2,$$

pero $(s \dot{-} 1)_1 = l = (t \dot{-} 1)_1$, luego de hecho $s \dot{-} 1 = t \dot{-} 1$, luego $s = t$. ∎

Ejemplo La tabla siguiente muestra la interpretación como sucesiones de los 40 primeros números naturales:

0		10	$0,0,0,0$	20	$0,0,0,0,1$	30	$0,3$
1	0	11	4	21	$0,0,0,0,0,0$	31	$1,0,0$
2	1	12	$0,2$	22	6	32	$0,0,1,1$
3	$0,0$	13	$0,1,0$	23	$2,0$	33	$0,0,0,0,2$
4	2	14	$0,0,0,1$	24	$0,1,1$	34	$0,0,0,0,1,0$
5	$0,1$	15	$0,0,0,0,0$	25	$0,0,0,2$	35	$0,0,0,0,0,0,1$
6	$0,0,0$	16	5	26	$0,0,0,1,0$	36	$0,0,0,0,0,0,0,0$
7	3	17	$1,1$	27	$0,0,0,0,0,1$	37	8
8	$1,0$	18	$0,0,2$	28	$0,0,0,0,0,0,0$	38	$1,2$
9	$0,0,1$	19	$0,0,1,0$	29	7	39	$0,0,3$

Por ejemplo, para calcular la sucesión asociada al número 352 889 465, como no es 0, le restamos 1 y lo interpretamos como par:

$$352\,889\,464 = \langle 3, 26\,563 \rangle_2,$$

lo que nos indica que debemos interpretar el número 26 563 como una sucesión de longitud 4, la cual resulta ser $3, 2, 2, 1$.

A partir de ahora, cuando hablemos de la sucesión $3, 2, 2, 1$, entenderemos que no es sino una forma de referirnos al número natural 352 889 465. ∎

Una comprobación rutinaria muestra que

$$p_i^\infty(\langle x_0, \ldots, x_{n-1} \rangle_\infty^n) = x_i,$$

lo que prueba que toda sucesión finita de números naturales está codificada por un número natural. En la práctica podemos omitir el superíndice n en el funtor $\langle x_0, \ldots, x_{n-1} \rangle_\infty^n$, puesto que éste se deduce del número de argumentos.

No debemos confundir un número natural n con la sucesión de longitud 1 que lo tiene como único término. Ésta viene dada por el funtor

$$\langle n \rangle_\infty = \langle 0, n \rangle_2 + 1.$$

Un último hecho general de interés sobre las proyecciones es el siguiente:

Teorema 2.7 $\ell(s) \leq s \ y \ \bigwedge i < \ell(s)\, s_i \leq s$.

DEMOSTRACIÓN: Claramente $\ell(s) = (s \dot{-} 1)+1 \leq s$. Para probar la segunda parte vemos primero por inducción que $s > 0 \to R(s,i) < s$. En efecto, para $i = 0$ es $R(s,0) = (s \dot{-} 1)_2 \leq s \dot{-} 1 < s$. Si es cierto para i, entonces

$$R(s, i+1) = R(s,i)_1 \leq R(s,i) < s.$$

Ahora probamos por inducción que $s > 0 \wedge i < \ell(s) \to s_i < s$. Para $i = 0$ es

$$s_0 = p_0^\infty(s) = R(s, (s \dot{-} 1)_1) < s.$$

Si vale para i y $i + 1 < \ell(s)$, entonces

$$s_{i+1} = p_{i+1}^\infty(s) = R(s, (s \dot{-} 1)_1 \dot{-} (i+1))_2 \leq R(s, (s \dot{-} 1)_1 \dot{-} (i+1)) < s.$$

Si $s = 0$ la conclusión es trivial. ∎

Definición 2.8 Diremos que una sucesión t *extiende* a otra s si

$$s \sqsubseteq t \equiv \ell(s) \leq \ell(t) \wedge \bigwedge i < \ell(s)\, s_i = t_i.$$

Es fácil ver que:

1. $s \sqsubseteq s$,
2. $s \sqsubseteq t \wedge t \sqsubseteq s \to s = t$,
3. $s \sqsubseteq t \wedge t \sqsubseteq u \to s \sqsubseteq u$,
4. $0 \sqsubseteq s$.

Definimos

$$s^\frown\langle n\rangle = (1 \dot{-} s)\, \langle n\rangle_\infty + (1 \dot{-} (1 \dot{-} s))(\langle \ell(s), \langle (s \dot{-} 1)_2, n\rangle_2\rangle_2 + 1).$$

Una comprobación rutinaria muestra que

$$\ell(s^\frown\langle n\rangle) = \ell(s) + 1, \quad s \sqsubseteq s^\frown\langle n\rangle\,, \quad (s^\frown\langle n\rangle)_{\ell(s)} = n.$$

Vemos así que $s^\frown\langle n\rangle$ no es sino la sucesión que resulta de añadir n como último término a la sucesión representada por s. Más en general, definimos

$$s^\frown t = F(s, t, \ell(t)),$$

donde F es el funtor dado por

$$F(s,t,0) = s, \qquad F(s,t,n+1) = F(s,t,n)^\frown\, \langle t_n\rangle\,.$$

De nuevo una comprobación rutinaria muestra que

$$\ell(s^\frown t) = \ell(s) + \ell(t), \quad \bigwedge i < \ell(s)(s^\frown t)_i = s_i, \quad \bigwedge i < \ell(t)\, (s^\frown t)_{\ell(s)+i} = t_i.$$

En particular, $s \sqsubseteq s^\frown t$.

Definimos la *restricción* de una sucesión mediante el funtor dado por

$$s|_0 = 0, \qquad s|_{i+1} = s|_i{}^\frown \langle s_i \rangle .$$

Es fácil probar que $i \le \ell(s) \to \ell(s|_i) = i \wedge s|_i \sqsubseteq s$. Más aún, si $t \sqsubseteq s$, entonces $t = s|_{\ell(t)}$.

Similarmente definimos $s|^i = F(s, i, \ell(s) \dot- i)$, donde

$$F(s,i,0) = s_i, \qquad F(s,i,n+1) = F(s,i,n)^\frown \langle s_{i+n} \rangle .$$

Así, si $i \le \ell(s)$, se cumple que $\ell(s|^i) = \ell(s) \dot- i$ y $s = s|_i{}^\frown s|^i$.

A partir de este momento ya no usaremos más los funtores $\langle x_1, \ldots, x_n \rangle_n$ ni sus proyecciones correspondientes, sino que siempre que hablemos de sucesiones de números naturales consideraremos los funtores $\langle x_1, \ldots, x_n \rangle_\infty$ y las proyecciones dadas por el funtor $p_i^\infty(s)$, de rango 2.

Si $F(x_1, \ldots, x_{n+1})$ es un funtor de rango $n+1$, podemos definir a partir de él otro funtor igualmente de rango $n+1$ dado por

$$F|_0(x_1, \ldots, x_n) = 0, \ F|_{x+1}(x_1, \ldots, x_n) = F|_x(x_1, \ldots, x_n)^\frown \langle F(x_1, \ldots, x_n, x) \rangle .$$

Es claro que $\ell(F|_x(x_1, \ldots, x_n)) = x$ y

$$\textstyle\bigwedge i < x \, F|_x(x_1, \ldots, x_n)_i = F(x_1, \ldots, x_n, i).$$

Más explícitamente:

$$F|_x(x_1, \ldots, x_n) = \langle F(x_1, \ldots, x_n, 0), \ldots, F(x_1, \ldots, x_n, n-1) \rangle_\infty .$$

Con esto podemos definir un funtor $F(x_1, \ldots, x_n, n)$ suponiendo definidos los valores $F(x_1, \ldots, x_n, 0), \ldots, F(x_1, \ldots, x_n, n-1)$. Más precisamente:

Teorema 2.9 (Recursión completa) *Si G es un funtor de rango $n+2$, existe un funtor F de rango $n+1$ tal que*

$$F(x_1, \ldots, x_n, x) = G(x_1, \ldots, x_n, x, F|_x(x_1, \ldots, x_n)).$$

Demostración: Definimos un funtor H mediante

$$\begin{aligned} H(x_1, \ldots, x_n, 0) &= 0, \\ H(x_1, \ldots, x_n, x+1) &= H(x_1, \ldots, x_n, x)^\frown \langle G(x_1, \ldots, x_n, x, H(x_1, \ldots, x_n, x)) \rangle . \end{aligned}$$

Y a su vez,

$$F(x_1, \ldots, x_n, x) = H(x_1, \ldots, x_n, x+1)_x.$$

Veamos que F cumple lo requerido. Por aligerar la notación escribiremos $\bar{x}$ en lugar de $x_1, \ldots, x_n$. En primer lugar, una simple inducción nos da que $\ell(H(\bar{x}, x)) = x$. En segundo lugar, $H(\bar{x}, x) = F|_x(\bar{x})$.

En efecto, para $x = 0$ es

$$H(\bar{x}, 0) = 0 = F|_0(\bar{x}),$$

y si vale para x, entonces, como $\ell(H(\bar{x}, x+1)) = x+1$, la definición de H implica que

$$H(\bar{x}, x+1) = H(\bar{x}, x)^\frown\langle H(\bar{x}, x+1)_x\rangle\,,$$

luego por la hipótesis de inducción y la definición de F:

$$H(\bar{x}, x+1) = F|_x(\bar{x})^\frown\langle F(\bar{x}, x)\rangle = F|_{x+1}(\bar{x}).$$

Finalmente probamos por inducción que F cumple la propiedad del enunciado:

$$F(\bar{x}, 0) = H(\bar{x}, 1)_0 = G(\bar{x}, 0, 0) = G(\bar{x}, 0, F|_0(\bar{x})),$$
$$F(\bar{x}, x+1) = H(\bar{x}, x+2)_{x+1} = G(\bar{x}, x+1, H(\bar{x}, x+1)) = G(\bar{x}, x+1, F|_{x+1}(\bar{x})).$$

■

Además, si dos funtores F_1 y F_2 cumplen el teorema anterior, podemos probar que

$$F_1(x_1, \ldots, x_n, x) = F_2(x_1, \ldots, x_n, x).$$

En efecto, supongamos que

$$F_1(x_1, \ldots, x_n, x) \neq F_2(x_1, \ldots, x_n, x).$$

Entonces podemos considerar

$$a \equiv \mu u \leq x\ F_1(x_1, \ldots, x_n, u) \neq F_2(x_1, \ldots, x_n, u)$$

y tenemos que

$$F_1(x_1, \ldots, x_n, a) \neq F_2(x_1, \ldots, x_n, a),$$
$$\bigwedge u < a F_1(x_1, \ldots, x_n, u) = F_2(x_1, \ldots, x_n, u)$$

de donde se sigue que $F_1|a(x_1, \ldots, x_n) = F_2|a(x_1, \ldots, x_n)$, pero entonces

$$F_1(x_1, \ldots, x_n, a) = G(x_1, \ldots, x_n, x, F_1|_a(x_1, \ldots, x_n)) =$$
$$G(x_1, \ldots, x_n, x, F_2|_a(x_1, \ldots, x_n)) = F_2(x_1, \ldots, x_n, a),$$

y tenemos una contradicción. ■

Por ejemplo, ahora podemos definir la sucesión de Fibonacci en ARP:

$$F(0) = F(1) = 1, \quad F(n) = F(n-2) + F(n-1), \quad n \geq 2.$$

Un poco más en general, dados tres términos t_0, t_1, t, siempre podemos definir un funtor por las condiciones siguientes:

$$F(\bar{x}, x) = \begin{cases} t_0(\bar{x}) & \text{si } x = 0, \\ t_1(\bar{x}) & \text{si } x = 1, \\ t(\bar{x}, x, F(\bar{x}, x \dot{-} 2), F(\bar{x}, x \dot{-} 1)) & \text{si } x \geq 2. \end{cases}$$

Para ajustarnos al teorema anterior definimos el funtor

$$G(\bar{x},x,s)=\begin{cases} t_0(\bar{x}) & \text{si } \ell(s)=0,\\ t_1(\bar{x}) & \text{si } \ell(s)=1,\\ t(\bar{x},x,s_{\ell(s)\dot{-}2},s_{\ell(s)\dot{-}1}) & \text{si } \ell(s)\geq 2,\end{cases}$$

o, equivalentemente,

$$G(\bar{x},x,s)=\chi_{u=v}(\ell(s),0)\cdot t_0(\bar{x})+\chi_{u=v}(\ell(s),1)\cdot t_1(\bar{x})$$

$$+\chi_{u\leq v}(2,\ell(s))\cdot t(\bar{x},x,s_{\ell(s)\dot{-}2},s_{\ell(s)\dot{-}1})$$

de modo que $F(\bar{x},x)=G(\bar{x},x,F|_x)$.

Terminamos este apartado con un resultado técnico que necesitaremos más adelante:

Teorema 2.10 $s\leq t^\frown s$.

DEMOSTRACIÓN: Si $t=0$ es obvio y no perdemos generalidad si suponemos que $t=\langle m\rangle$ tiene longitud 1. Sea $n\geq 1$ y supongamos que $\ell(s)=n+1$. Definimos

$$s^*=\langle n\dot{-}1,R(s,1)\rangle_2+1$$

donde R es el funtor definido en 2.5. Vamos a probar por inducción que

$$R(s^*,i)=R(s,i+1).$$

Para $i=0$ tenemos $R(s^*,0)=(s^*\dot{-}1)_2=R(s,1)$. Supuesto cierto para i,

$$R(s^*,i+1)=R(s^*,i)_1=R(s,i+1)_1=R(s,i+2).$$

Ahora veamos que $\bigwedge i<n s_i^*=s_i$. Para $i=0$ tenemos:

$$s_0^*=p_0^\infty(s^*)=R(s^*,(s^*\dot{-}1)_1)=R(s^*,n\dot{-}1)$$

$$=R(s,n)=R(s,(s\dot{-}1)_1)=p_0^\infty(s)=s_0.$$

Si vale para i y tenemos que $i+1<n$, entonces

$$s_{i+1}^*=p_{i+1}^\infty(s^*)=R(s^*,(s^*\dot{-}1)_1\dot{-}(i+1))=R(s,n\dot{-}(i+1))$$

$$=R(s,(s\dot{-}1)_1\dot{-}(i+1))=p_{i+1}^\infty(s)=s_{i+1}.$$

Como $\ell(s^*)=n$, hemos probado que $\bigwedge i<\ell(s^*)\ s_i^*=s_i$, luego $s^*=s|_n$.

Volvemos ahora a la igualdad $R(s*,0)=R(s,1)=R(s,0)_1$, que ahora se convierte en $R(s|_n,0)=R(s,0)_1$. Por otra parte,

$$s_n=p_n^\infty(s)=R(s,n\dot{-}n)_2=R(s,0)_2.$$

Así pues, como $R(s,0)_1=R(s|_n,0)$ y $R(s,0)_2=s_n$, tenemos que

$$R(s,0)=\langle R(s|_n,0),s_n\rangle_2$$

siempre que $\ell(s) = n + 1 \geq 2$. Más en general, si $\ell(s) \geq n + 1 \geq 2$, aplicando esto a $s|_{n+1}$, concluimos que

$$R(s|_{n+1}, 0) = \langle R(s|_n, 0), s_n \rangle_2 .$$

Ahora vamos a probar que si $1 \leq n \leq \ell(s)$, se cumple

$$R(s|_n, 0) \leq R(\langle m \rangle \frown s|_n, 0).$$

Por inducción sobre n. Para $n = 1$ se cumple que

$$s|_1 = \langle 0, s_0 \rangle_2 + 1, \qquad \langle m \rangle \frown s|_1 = \langle 1, \langle m, s_0 \rangle_2 \rangle_2 + 1,$$

luego $R(s|_1, 0) = s_0 \leq \langle m, s_0 \rangle_2 = R(\langle m \rangle \frown s|_1, 0)$.

Si vale para $n \geq 1$, entonces

$$R(s|_{n+1}, 0) = \langle R(s|_n, 0), s_n \rangle_2 \leq \langle R(\langle m \rangle \frown s|_n, 0), s_n \rangle_2$$

$$= \langle R((\langle m \rangle \frown s)|_{n+1}, 0), s_n \rangle_2 = R((\langle m \rangle \frown s)|_{n+2}, 0) = R(\langle m \rangle \frown s|_{n+1}, 0).$$

En particular, para $n = \ell(s) \geq 1$, tenemos que $R(s, 0) \leq R(\langle m \rangle \frown s, 0)$, y a su vez

$$s = \langle n \dot{-} 1, R(s, 0) \rangle_2 + 1 \leq \langle n, R(\langle m \rangle \frown s, 0) \rangle_2 + 1 = \langle m \rangle \frown s.$$

■

Sumas finitas Ahora que ya sabemos manipular sucesiones finitas en ARP, pasamos a estudiar las sumas finitas:

Definición 2.11 Si $t(x_1, \ldots, x_n)$ es un término cuyas variables estén entre las indicadas, definimos un funtor de rango n mediante las ecuaciones:

$$\begin{aligned} \sum_{i<0} t(x_1, \ldots, x_{n-1}, i) &= 0 \\ \sum_{i<x+1} t(x_1, \ldots, x_{n-1}, i) &= \sum_{i<x} t(x_1, \ldots, x_{n-1}, i) + t(x_1, \ldots, x_{n-1}, x). \end{aligned}$$

Escribiremos también:

$$\sum_{i \leq x} t(x_1, \ldots, x_{n-1}, i) \equiv \sum_{i<x+1} t(x_1, \ldots, x_{n-1}, i).$$

Los hechos siguientes se demuestran fácilmente por inducción sobre x (por simplicidad omitimos los parámetros que no son relevantes en el enunciado de las propiedades):

1. $\sum_{i<x} t_1(i) + \sum_{i<x} t_2(i) = \sum_{i<x} (t_1(i) + t_2(i))$.
2. $\sum_{i<x} y \cdot t(i) = y \sum_{i<x} t(i)$.
3. $\sum_{i<y+x} t(i) = \sum_{i<y} t(i) + \sum_{i<x} t(y + i)$.
4. $\bigwedge i < x \; t_1(i) \leq t_2(i) \to \sum_{i<x} t_1(i) \leq \sum_{i<x} t_2(i)$.
5. $\sum_{i<x} t(i) = 0 \to \bigwedge i < x \; t(i) = 0$.

Desarrollos decimales Hasta ahora, cuando hemos escrito números como 184, teníamos que entender que 184 no era más que una forma de abreviar un numeral compuesto por 184 funtores S seguidos de un 0. Sin embargo ahora podemos formalizar en ARP el concepto de desarrollo decimal que nos permite interpretar 184 como lo que expresa realmente. Empezamos introduciendo las definiciones necesarias:

Definición 2.12 Consideremos el funtor dado por

$$s_{(d)} = \sum_{i<\ell(s)} s_i d^i.$$

Un *desarrollo decimal en base d* de un número natural x es una sucesión finita s tal que $\bigwedge i < \ell(s)\, s_i < d$ y $x = s_{(d)}$. Diremos que se trata de un desarrollo *reducido* si $\ell(s) = 0$ (lo cual sólo puede suceder si $x = 0$) o bien $\ell(s) > 0$ y $s_{\ell(s)\dot{-}1} \neq 0$.

Vamos a demostrar que cada número natural admite un desarrollo decimal en cualquier base $d \geq 2$ y que éste es único si exigimos que sea reducido.

Empezamos considerando el funtor dado por las ecuaciones

$$\begin{aligned} F(x,d,0) &= \langle c(x,d), r(x,d)\rangle\,, \\ F(x,d,i+1) &= \langle c(F(x,d,i)_0,d), r(F(x,d,i)_0,d)\rangle\,. \end{aligned}$$

donde c y r son los funtores que calculan el cociente y el resto de la división euclídea. Definimos $x_i^*[d] = F(x,d,i)_0$, $x_i[d] = F(x,d,i)_1$. Así

$$x = x_0^*[d]\cdot d + x_0[d], \qquad x_i^*[d] = x_{i+1}^*[d]\cdot d + x_{i+1}[d],$$

con $x_i[d] < d$, para todo i. De aquí se sigue que

$$x = x_n^*[d]\cdot d^{n+1} + \sum_{i\leq n} x_i[d]\cdot d^i.$$

En efecto, razonamos por inducción sobre n. Para $n = 0$ es inmediato y, si vale para n,

$$\begin{aligned} x &= x_n^*[d]\cdot d^{n+1} + \sum_{i\leq n} x_i[d]\cdot d^i \\ &= (x_{n+1}^*[d]\cdot d + x_{n+1}[d])\cdot d^{n+1} + \sum_{i\leq n} x_i[d]\cdot d^i \\ &= x_{n+1}^*[d]\cdot d^{n+2} + x_{n+1}[d]\cdot d^{n+1} + \sum_{i\leq n} x_i[d]\cdot d^i \\ &= x_{n+1}^*[d]\cdot d^{n+2} + \sum_{i\leq n+1} x_i[d]\cdot d^i. \end{aligned}$$

Si $d \geq 2$ y $n \geq x$, tenemos que $x < d^{x+1} \leq d^{n+1}$, luego $x_n^*[d] = 0$, pues de lo contrario la expresión que acabamos de probar nos daría que $d^{n+1} \leq x$. Por consiguiente,

$$x = \sum_{i\leq n} x_i[d]\cdot d^i.$$

Con esto ya hemos probado que todo número natural admite un desarrollo decimal en base d. Vamos a afinar un poco la construcción para quedarnos con uno reducido. Definimos

$$N_d(x) = \mu u \leq x+1\ x = \sum_{i<u} x_i[d]\cdot d^i,$$

de modo que

$$x = \sum_{i<N_d(x)} x_i[d]\cdot d^i.$$

Claramente $N_d(0) = 0$ y si $x \neq 0$, entonces $N_d(x) > 0$ y $n_d(x) = N_d(x) \dot{-} 1$ cumple que $x_{n_d(x)}[d] \neq 0$, o de lo contrario sería

$$x = \sum_{i<n_d(x)} x_i[d]\cdot d^i$$

con $n_d(x) < N_d(x)$, en contra de la minimalidad de $N_d(x)$.

Definimos las *cifras decimales de x en base d* como la sucesión dada por la restricción a $N_d(x)$ del funtor $x_i[d]$, es decir, $x[d] = x|_{N_d(x)}[d]$. El teorema siguiente recoge lo que hemos demostrado y algunos hechos más:

Teorema 2.13 *Si $d \geq 2$, todo número natural $x \neq 0$ admite el desarrollo decimal reducido en base d*

$$x = x[d]_{(d)} = \sum_{i\leq n_d(x)} x_i[d]\cdot d^i.$$

Además, $x_i[d] = 0$ para todo $i > n_d(x)$ y si $x = s_{(d)}$ es un desarrollo decimal de x en base d, necesariamente $\ell(s) > n_d(x)$ y $\bigwedge i < \ell(s)\, s_i = x_i[d]$.

Demostración: Ya hemos probado que $x[d]_{(d)} = x$, que $x_i[d] < d$ y que $x_{n_d(x)}[d] \neq 0$, por lo que $x[d]$ es un desarrollo decimal reducido de x.

Para probar la unicidad partimos de una sucesión $s > 0$ arbitraria tal que $\bigwedge i < \ell(s)\, s_i < d$ y vamos a probar por inducción que, para todo $k < \ell(s)$, se cumple

$$(s_{(d)})^*_k[d] = \sum_{i<\ell(s)\dot{-}(k+1)} s_{i+k+1}d^i \wedge \bigwedge i \leq k\, (s_{(d)})_i[d] = s_i.$$

Para $k = 0$ tenemos que

$$s_{(d)} = \sum_{i<\ell(s)} s_i\cdot d^i = \sum_{i<\ell(s)\dot{-}1} s_{i+1}\cdot d^{i+1} + s_0 = d\cdot \sum_{i<\ell(s)\dot{-}1} s_{i+1}\cdot d^i + s_0,$$

luego por la unicidad de la división euclídea concluimos que $(s_{(d)})_0[d] = s_0$ y

$$(s_{(d)})^*_0[d] = \sum_{i<\ell(s)\dot{-}1} s_{i+1}\cdot d^i.$$

Supuesto cierto para k y que $k+1 < \ell(s)$, tenemos que

$$(s_{(d)})^*_k[d] = d \cdot (s_{(d)})^*_{k+1}[d] + (s_{(d)})_{k+1}[d], \qquad (s_{(d)})_{k+1}[d] < d,$$

$$(s_{(d)})^*_k[d] = \sum_{i<\ell(s)\dot{-}(k+1)} s_{i+k+1}d^i = d \cdot \sum_{i<\ell(s)\dot{-}(k+2)} s_{i+k+2}d^i + s_{k+1}, \qquad s_{k+1} < d.$$

Por la unicidad de la división euclídea, tiene que ser

$$(s_{(d)})^*_{k+1}[d] = \sum_{i<\ell(s)\dot{-}(k+2)} s_{i+k+2}d^i, \qquad (s_{(d)})_{k+1}[d] = s_{k+1},$$

lo que completa la inducción. En particular, para $k = \ell(s) \dot{-} 1$ tenemos que

$$\bigwedge i < \ell(s)\, (s_{(d)})_i[d] = s_i.$$

Si suponemos que $s_{(d)} = x$, entonces $x = \sum_{i<\ell(s)} x_i[d] \cdot d^i$, luego por la definición de $N_d(x)$ tiene que ser $n_d(x) < N_d(x) \le \ell(s)$. Además,

$$x = \sum_{i<\ell(s)} s_i \cdot d^i = \sum_{i<N_d(x)} x_i[d] \cdot d^i + \sum_{i<\ell(s)\dot{-}N_d(x)} s_{N_d(x)+i}\, d^{N_d(x)+i},$$

pero el primero de los últimos dos sumandos también vale x, luego

$$\sum_{i<\ell(s)\dot{-}N_d(x)} s_{N_d(x)+i}\, d^{N_d(x)+i} = 0,$$

luego $\bigwedge i < \ell(s) \dot{-} N_d(x)\ s_{N_d(x)+i} = 0$, luego $s_i = 0$ si $N_d(x) \le i < \ell(s)$.

Esto se aplica a los desarrollos

$$x = \sum_{i\le n} x_i[d] \cdot d^i,$$

válidos para $n \ge x$, lo que prueba que $x_i[d] = 0$ siempre que $i > n_d(x)$ y así hemos probado que $\bigwedge i < \ell(s) s_i = x_i[d]$. ∎

Por otro lado, observemos que $0 = 0_{(d)}$ es por definición el único desarrollo decimal reducido de 0.

En particular tenemos que un número natural está determinado por sus cifras decimales:

Teorema 2.14 $d \ge 2 \wedge \bigwedge i < x + y\ x_i[d] = y_i[d] \to x = y$.

Así pues, ahora podemos definir las cifras decimales

$$1 \equiv S0, \quad 2 \equiv S1, \quad 3 \equiv S2, \quad 4 \equiv S3, \quad 5 \equiv S4,$$

$$6 \equiv S5, \quad 7 \equiv S6, \quad 8 \equiv S7, \quad 9 \equiv S8$$

y convenir en que todo numeral, como 1 314, ha de interpretarse como

$$\langle 4, 1, 3, 1\rangle_{(S9)}\,.$$

En particular, $S9 = 10$ y $\langle 4, 1, 3, 1\rangle_{(10)} = 1 \cdot 10^3 + 3 \cdot 10^2 + 1 \cdot 10 + 4$.

2.2 Conjuntos finitos

Ahora veremos que, al igual que podemos interpretar los números naturales como sucesiones finitas, también podemos interpretarlos como conjuntos finitos. Para ello basta considerar la relación de pertenencia siguiente:

Definición 2.15 $x \in y \equiv y_x[2] = 1, \qquad x \notin y \equiv \neg x \in y.$

Equivalentemente, un número natural x pertenece a otro y si la cifra de orden x del desarrollo binario de y vale 1.

En particular, si $x \in y$, entonces $x < 2^x \leq y$. Esto hace que podamos considerar cuantificadores acotados en la forma

$$\bigvee u \in y\, \alpha(u) \equiv \bigvee u \leq y(u \in y \wedge \alpha(u)), \quad \bigwedge u \in y\, \alpha(u) \equiv \bigwedge u \leq y(u \in y \to \alpha(u)),$$

$$\overset{1}{\bigvee} u \in y\, \alpha(u) \equiv \overset{1}{\bigvee} u \leq y(u \in y \wedge \alpha(u)).$$

La unicidad de los desarrollos decimales implica claramente lo que en teoría de conjuntos se llama axioma de extensionalidad:

$$\bigwedge u \in x\, u \in y \wedge \bigwedge u \in y\, u \in x \to x = y.$$

Definimos la inclusión como

$$x \subset y \equiv \bigwedge u \in x\, u \in y,$$

de modo que

1. $x \subset x$,
2. $x \subset y \wedge y \subset x \to x = y$,
3. $x \subset y \wedge y \subset z \to x \subset z$.

Conviene observar que se cumple también:

$$x \subset y \to x \leq y.$$

En efecto, $x \subset y \to x_i[2] \leq y_i[s]$, de donde $x \leq y$.

Definimos $\varnothing \equiv 0$, y es inmediato que $x \notin \varnothing$, así como que $\varnothing \subset x$.

Dada una fórmula $\alpha(u, x_1, \ldots, x_n)$, definimos

$$\{u < k \mid \alpha(u, x_1, \ldots, x_n)\} \equiv \sum_{u<k} \chi_\alpha(u, x_1, \ldots, x_n) 2^u,$$

y así se cumple claramente lo que en teoría de conjuntos se conoce como *axioma de especificación:*

Teorema 2.16 (Especificación) *Si $\alpha(x, x_1, \ldots, x_n)$ es una fórmula cualquiera, en* ARP *se demuestra:*

$$x \in \{u < k \mid \alpha(u, x_1, \ldots, x_n)\} \leftrightarrow x < k \wedge \alpha(x, x_1, \ldots, x_n).$$

Notemos que en el teorema anterior podemos cambiar $u < k$ por $u \in k$, entendiendo que

$$\{u \in k \mid \alpha(u, x_1, \ldots, x_n)\} \equiv \{u < k \mid u \in k \wedge \alpha(u, x_1, \ldots, x_n)\}.$$

A partir de aquí ya podemos definir fácilmente todos los conceptos conjuntistas básicos:

$$\begin{aligned}
\{x_1, \ldots, x_n\} &\equiv \{u < x_1 + \cdots + x_n \mid u = x_1 \vee \cdots \vee u = x_n\},\\
x \cup y &\equiv \{u < x + y \mid u \in x \vee u \in y\},\\
x \cap y &\equiv \{u < x + y \mid u \in x \wedge u \in y\},\\
x \setminus y &\equiv \{u < x \mid u \in x \wedge u \notin y\},\\
\mathcal{P}x &\equiv \{u < x + 1 \mid u \subset x\},\\
\bigcup x &\equiv \{u < x \mid \bigvee v \in x\, u \in v\},\\
\bigcap x &\equiv \{u < x \mid \bigwedge v \in x\, u \in v\},\\
x \times y &\equiv \{w < \langle x, y\rangle \mid \bigvee u \in x \bigvee v \in y\, w = \langle u, v\rangle\}.
\end{aligned}$$

Observemos que

$$x \neq \varnothing \to (u \in \bigcap x \leftrightarrow \bigwedge v \in x\, u \in v),$$

mientras que $\bigcap \varnothing = \varnothing$ no cumple la equivalencia anterior.

Definimos también $I_k \equiv \{u < k \mid u = u\}$, que es el conjunto de todos los números naturales menores que k.

Nota: Pares ordenados Con la definición que hemos dado de producto cartesiano $x \times y$ se cumple que

$$w \in x \times y \leftrightarrow \bigvee u \in x \bigvee v \in y\ w = \langle u, v\rangle\,.$$

En teoría de conjuntos es habitual definir el par ordenado de dos conjuntos como

$$(x, y) \equiv \{\{x\}, \{x, y\}\}.$$

Esta definición también es válida en ARP y se demuestra fácilmente que

$$(x, y) = (z, w) \leftrightarrow x = z \wedge y = w.$$

También se cumple que $x < \{x\} < (x, y)$, $y < \{x, y\} < (x, y)$. Por consiguiente podríamos haber definido

$$x \times y \equiv \{w < 2^{2^x} + 2^{2^x + 2^y} \mid \bigvee u \in x \bigvee v \in y\ w = (u, v)\}.$$

y se cumpliría igualmente

$$w \in x \times y \leftrightarrow \bigvee u \in x \bigvee v \in y\ w = (u, v).$$

Esto hace que, en todo cuanto vamos a decir a continuación sobre aplicaciones (que las vamos a definir como subconjuntos de productos cartesianos) será irrelevante considerar pares ordenados de la forma $\langle u, v\rangle$ o (u, v). ∎

Para definir subconjuntos de productos cartesianos es útil introducir la notación siguiente:

$$\{\langle u, v\rangle \in x \times y \mid \alpha(u, v, x, y, x_1, \ldots, x_n)\} \equiv$$

$$\{w \in x \times y \mid \bigvee u \in x \bigvee v \in y(w = \langle u, v\rangle \wedge \alpha(u, v, x, y, x_1, \ldots, x_n))\}.$$

El mínimo y el máximo de un conjunto pueden definirse así:

$$\text{mín}\, x = \mu u \leq x(u \in x), \qquad \text{máx}\, x = x \dot{-} \mu u \leq x(x \dot{-} u \in x),$$

de modo que

$$x \neq \varnothing \to (\text{mín}\, x \in x \wedge \bigwedge u \in x\ \text{mín}\, x \leq u),$$
$$x \neq \varnothing \to (\text{máx}\, x \in x \wedge \bigwedge u \in x\, u \leq \text{máx}\, x).$$

Veamos ahora que es posible definir conjuntos sin tener a priori una cota para sus elementos eludiendo la exigencia del teorema 2.16.

Si fijamos un término $t(x, x_1, \ldots, x_n)$, podemos definir el funtor

$$\begin{aligned} F_t(x_1, \ldots, x_n, y, 0) &= 0, \\ F_t(x_1, \ldots, x_n, y, x+1) &= F_t(x_1, \ldots, x_n, y, x) \cup \{t(x, x_1, \ldots, x_n)\} \cdot \chi[x \in y], \end{aligned}$$

y a su vez

$$\{t(u, x_1, \ldots, x_n) \mid u \in y\} = F_t(x_1, \ldots, x_n, y, y).$$

Teorema 2.17 (Reemplazo) *Si $t(x, x_1, \ldots, x_n)$ es un término cualquiera, en* ARP *se demuestra:*

$$z \in \{t(u, x_1, \ldots, x_n) \mid u \in y\} \leftrightarrow \bigvee u \in y\, z = t(u, x_1, \ldots, x_n).$$

Demostración: Omitiendo los parámetros, supongamos que

$$z \in \{t(u) \mid u \in y\} = F_t(y, y).$$

Podemos tomar entonces $w \equiv \mu u \leq y\, z \in F_t(y, u)$. No puede ser $w = 0$, luego, llamando $x = w \dot{-} 1$, se cumple que $w = x + 1$ y así

$$z \in F_t(y, x+1) = F_t(y, x) \cup \{t(x)\} \cdot \chi(x \in y).$$

Por la minimalidad de w, tenemos que $z \notin F_t(y, x)$, luego se tiene que cumplir que $z = t(x)$ y $\chi[x \in y] = 1$, que es lo mismo que $x \in y$. Por lo tanto, $\bigvee u \in y\, z = t(u)$.

Recíprocamente, supongamos que $\bigvee u \in y\, z = t(u)$ y tomemos

$$x \equiv \mu u \in y\, z = t(u).$$

Entonces $x \in y$, luego $x < y$, luego $x + 1 \leq y$. Además, $z = t(x)$, luego

$$z \in F_t(y, x) \cup \{t(x)\} \cdot \chi[x \in y] = F_t(y, x + 1).$$

Ahora bien, una simple inducción demuestra que

$$x_1 \leq x_2 \to F_t(y, x_1) \subset F_t(y, x_2),$$

luego podemos concluir que $z \in F_t(y, y) = \{t(u) \mid u \in y\}$. ∎

En otras palabras, los valores que toma un término cuando una de sus variables recorre un conjunto, forman un conjunto.

Por ejemplo, ahora podemos definir

$$\bigcup_{u\in y} t(u, x_1, \ldots, x_n) = \bigcup\{t(u, x_1, \ldots, x_n) \mid u \in y\},$$

$$\bigcap_{u\in y} t(u, x_1, \ldots, x_n) = \bigcap\{t(u, x_1, \ldots, x_n) \mid u \in y\},$$

de modo que

$$v \in \bigcup_{u\in y} t(u, x_1, \ldots, x_n) \leftrightarrow \bigvee u \in y\, v \in t(u, x_1, \ldots, x_n),$$

$$y \neq \varnothing \to (v \in \bigcap_{u\in y} t(u, x_1, \ldots, x_n) \leftrightarrow \bigwedge u \in y\, v \in t(u, x_1, \ldots, x_n)).$$

Clausuras transitivas Un conjunto se dice *transitivo* si cumple

$$x \text{ transitivo } \equiv \bigwedge v \in x\, v \subset x$$

Más adelante necesitaremos usar que todo conjunto está contenido en un conjunto transitivo. De hecho, podemos asociar a cada conjunto un mínimo conjunto transitivo en el que está contenido, lo que se llama su clausura transitiva. Para definirlo definimos primero un funtor

$$G(x, s, 0) = \varnothing, \qquad G(x, s, i + 1) = \begin{cases} G(x, s, y) \cup s_y & \text{si } y \in x, \\ G(x, s, y) & \text{si } y \notin x. \end{cases}$$

A su vez, definimos $G(x, s) = x \cup G(x, s, \ell(s))$. Es claro entonces que

$$u \in G(x, s) \leftrightarrow u \in x \vee \bigvee y < \ell(s)(y \in x \wedge u \in s_y).$$

Usamos este funtor para definir la *clausura transitiva* por recursión completa:

$$\mathrm{ct}(x) = G(x, \mathrm{ct}|_x).$$

Explícitamente (teniendo en cuenta que $\ell(\mathrm{ct}|_x) = x$ y que $y \in x \to y < x$):

$$u \in \mathrm{ct}(x) \leftrightarrow u \in x \vee \bigvee y \in x\, u \in \mathrm{ct}(y).$$

Podemos expresar esto así:

$$\mathrm{ct}(x) = x \cup \bigcup_{y \in x} \mathrm{ct}(y).$$

Por ejemplo

$$\mathrm{ct}(0) = \varnothing, \quad \mathrm{ct}(1) = \{0\} \cup \varnothing = \{0\}, \quad \mathrm{ct}(2) = \{1\} \cup \{0\} = \{0,1\},$$

$$\mathrm{ct}(3) = \{0,1\} \cup \varnothing \cup \{0\} = \{0,1\}, \quad \mathrm{ct}(4) = \{2\} \cup \{0,1\} = \{0,1,2\}.$$

Los hechos básicos sobre clausuras transitivas son los siguientes:

Teorema 2.18 *Se cumple:*

1. $x \subset \mathrm{ct}(x) \wedge \mathrm{ct}(x)$ *es un conjunto transitivo.*
2. *Si* z *es un conjunto transitivo y* $x \subset z$*, entonces* $\mathrm{ct}(x) \subset z$*.*
3. $\mathrm{ct}(x) \cup \{x\}$ *es un conjunto transitivo.*

Demostración: 1) Obviamente $x \subset \mathrm{ct}(x)$. Probamos que ct es transitivo por inducción completa sobre x. Si $v \in \mathrm{ct}(x)$, o bien $v \in x$, o bien existe un $y \in x$ tal que $v \in \mathrm{ct}(y)$. En el primer caso $v \subset \mathrm{ct}(v) \subset \mathrm{ct}(x)$, mientras que en el segundo, por hipótesis de inducción $v \subset \mathrm{ct}(y) \subset \mathrm{ct}(x)$, luego en ambos casos $v \subset \mathrm{ct}(x)$.

2) Lo razonamos por inducción completa sobre x. Si es cierto para todo $y < x$ y $x \subset z$, entonces, si $y \in x$, se cumple que $y \in z$, luego $y \subset z$ y, por hipótesis de inducción, $\mathrm{ct}(y) \subset z$. Es claro entonces que $\mathrm{ct}(x) \subset z$.

3) se comprueba sin dificultad. ∎

Notemos que 2) expresa que $\mathrm{ct}(x)$ es el menor conjunto transitivo que contiene a x, mientras que es fácil ver que $\mathrm{ct}(x) \cup \{x\}$ es el menor conjunto transitivo al que pertenece x.

Aplicaciones Aunque las definiciones siguientes tienen sentido para cualquier conjunto f (es decir, para cualquier número natural f), sólo tienen interés cuando f es un conjunto de pares ordenados.

Definición 2.19 El *dominio* de un conjunto f es el conjunto de todas las primeras componentes de los pares ordenados de f y el *rango* es el conjunto de todas sus segundas componentes:

$$\mathcal{D}f \equiv \{u < f \mid \bigvee v < f\ \langle u, v\rangle \in f\},$$

$$\mathcal{R}f \equiv \{v < f \mid \bigvee u < f\ \langle u, v\rangle \in f\}.$$

Una *aplicación* f de un conjunto x en un conjunto y es un subconjunto $f \subset x \times y$ de modo que, para cada $u \in x$, haya un único par $\langle u, v\rangle \in f$. El elemento v se llama *imagen* de u por la aplicación f, y se representa por $f(u)$. También se dice que u es una *antiimagen* de v. Técnicamente:

$$f : x \longrightarrow y \equiv f \subset x \times y \wedge \bigwedge u \in f \overset{1}{\bigvee} v \in y \ \langle u, v\rangle \in f.$$

Así, si definimos

$$f(u) \equiv \mu v \in \mathcal{R}f \ \ \langle u, v\rangle \in f,$$

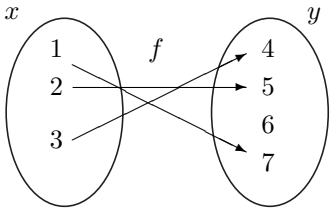
tenemos que

$$f : x \longrightarrow y \wedge u \in x \to f(u) \in y \wedge \langle u, f(u)\rangle \in f,$$

y también que

$$f : x \longrightarrow y \wedge \langle u, v\rangle \in f \to v = f(u).$$

Podemos representar los pares que componen una aplicación f como flechas que unen cada elemento de x con su imagen en y, así:



Notemos que

$$f : x \longrightarrow y \wedge g : x \longrightarrow z \wedge \bigwedge u \in x \ f(u) = g(u) \to f = g.$$

Podemos definir conjuntos de imágenes y antiimágenes:

$$\begin{aligned} f[z] &\equiv \{v \in \mathcal{R}f \mid \bigvee u \in z \, f(u) = v\}, \\ f^{-1}[z] &\equiv \{u \in \mathcal{D}f \mid f(u) \in z\}. \end{aligned}$$

Cualquier aplicación se puede restringir a un subconjunto de su dominio, así:

$$f|_z \equiv f \cap (z \times \mathcal{R}f).$$

Es fácil probar que

$$f : x \longrightarrow y \wedge z \subset x \to f|_z : z \longrightarrow y \wedge \bigwedge u \in z \, f|_z(u) = f(u).$$

Definimos como sigue la *composición* de dos aplicaciones f y g:

$$f \circ g \equiv \{\langle u, w\rangle \in \mathcal{D}f \times \mathcal{R}g \mid \bigvee v \in \mathcal{D}g \ (\langle u, v\rangle \in f \wedge \langle v, w\rangle \in g)\}.$$

De este modo:

$$f : x \longrightarrow y \wedge g : y \longrightarrow z \to f \circ g : x \longrightarrow z$$

y

$$f : x \longrightarrow y \wedge g : y \longrightarrow z \wedge u \in x \to (f \circ g)(u) = g(f(u)),$$

con lo que $f \circ g$ es la aplicación que se calcula aplicando f y luego aplicando g a la imagen obtenida.

Una aplicación es *inyectiva* si asigna imágenes distintas a elementos distintos y es *suprayectiva* si todo elemento del conjunto final tiene una antiimagen:

$$\begin{aligned} f : x \longrightarrow y \text{ inyectiva} &\equiv f : x \longrightarrow y \wedge \bigwedge uv \in \mathcal{D}f(f(u) = f(v) \to u = v),\\ f : x \longrightarrow y \text{ suprayectiva} &\equiv f : x \longrightarrow y \wedge \bigwedge v \in y \bigvee u \in x\, f(x) = y. \end{aligned}$$

Equivalentemente, f es suprayectiva cuando $f[x] = y$. Diremos que f es *biyectiva* si es inyectiva y suprayectiva a la vez, lo cual equivale a que f hace corresponder cada elemento del conjunto inicial con un único elemento del conjunto final, y viceversa. Esto hace que si definimos

$$f^{-1} \equiv \{w \in \mathcal{R}f \times \mathcal{D}f \mid \bigvee uv < w(w = \langle v, u\rangle \wedge \langle u, v\rangle \in f)\},$$

se cumpla que

$$f : x \longrightarrow y \text{ biyectiva} \to f^{-1} : y \longrightarrow x \text{ biyectiva},$$

$$f : x \longrightarrow y \text{ biyectiva} \wedge u \in x \wedge v \in y \to (f(u) = v \leftrightarrow f^{-1}(v) = u).$$

Por último definimos:

$$i_x \equiv \{\langle u, v\rangle \in x \times x \mid v = u\},$$

de modo que si $x \subset y$, se cumple que $i_x : x \longrightarrow y$ inyectiva. Esta aplicación se llama *inclusión* de x en y, mientras que si $x = y$ se llama *aplicación identidad* en x.

Teniendo en cuenta que

$$f : x \longrightarrow y \to f \in \mathcal{P}(x \times y),$$

podemos definir el *conjunto potencia*

$$y^x \equiv \{f \in \mathcal{P}(x \times y) \mid f : x \longrightarrow y\},$$

de modo que

$$f \in y^x \leftrightarrow f : x \longrightarrow y.$$

Cardinales Diremos que dos conjuntos son *equipotentes* si cumplen

$$x \sim y \equiv \bigvee f \in y^x\ f : x \longrightarrow y \text{ biyectiva}.$$

Esto significa que podemos emparejar cada elemento de x con un elemento de y, lo cual es posible si y sólo si x e y tienen el mismo número de elementos. Es fácil comprobar que

$$x \sim x, \qquad x \sim y \to y \sim x, \qquad x \sim y \wedge y \sim z \to x \sim z.$$

Para determinar el número de elementos de un conjunto x, lo que hacemos es *contar,* que no es sino establecer una biyección entre los elementos de x y un conjunto de números naturales de la forma $\{1, \ldots, k\}$. Decimos entonces que x tiene k elementos. Por razones técnicas, es más conveniente contar estableciendo biyecciones con conjuntos $I_k = \{0, \ldots, k-1\}$. Veamos que siempre es posible:

Teorema 2.20 $\bigvee k \leq x \bigvee f \in x^{I_k}\ f : I_k \longrightarrow x$ *biyectiva.*

DEMOSTRACIÓN: Lo probamos por inducción sobre x. Si $x = 0$ es obvio que sirve $k = 0$, pues $\varnothing : \varnothing \longrightarrow \varnothing$ biyectiva. Si es cierto para todo $y < x$ y $x \neq 0$, tomamos $u \in x$, con lo que $y = x \setminus \{u\} < x$. Por hipótesis de inducción existe un $k \leq y$ y una biyección $f : I_k \longrightarrow y$, que se extiende a $g : I_{k+1} \longrightarrow x$ biyectiva sin más que tomar $g = f \cup \{\langle k, u\rangle\}$. Además $k + 1 \leq y + 1 \leq x$. ∎

Ahora necesitamos observar que un mismo conjunto x no puede ser equipotente a dos conjuntos I_k, I_r con $k \neq r$. Si así fuera, serían $I_k \sim I_r$, luego basta ver que esto es imposible:

Teorema 2.21 $k \neq r \to I_k \not\sim I_r$.

DEMOSTRACIÓN: Basta probar que $k < r \to I_k \not\sim I_r$. Razonamos por inducción sobre r. Si $r = 0$ es trivial. Si vale para r y $k < r + 1$, pero existe $f : I_k \longrightarrow I_{r+1}$ biyectiva, sea $u < k$ tal que $f(u) = r$. En particular, $k \neq 0$, luego, llamando $k' = k \dot{-} 1 < r$, tenemos que $k = k' + 1$ y podemos construir $g : I_{k'} \longrightarrow I_r$ biyectiva como sigue: si $u = k'$ basta tomar $g = f|_{I_{k'}}$, mientras que si $u < k'$, entonces $f(k') \neq f(u) = r$, luego podemos definir $g(u) = f(k')$ y $g(v) = f(v)$ para todo $v < k'$, $v \neq u$. Esto contradice la hipótesis de inducción. ∎

Con esto podemos definir el *cardinal* de un conjunto x como

$$|x| = \mu k \leq x\ I_k \sim x.$$

Por comodidad lo hemos definido como "el mínimo k", pero en realidad hemos demostrado que es el único. Así:

$$I_{|x|} \sim x, \qquad I_k \sim x \to |x| = k.$$

El segundo teorema expresa que si hemos logrado biyectar un conjunto x con un conjunto I_k, entonces hemos contado x y podemos afirmar que su número de elementos es k.

En efecto, si $I_k \sim x$, entonces $I_{|x|} \sim x \sim I_k$, y hemos probado que no puede ser $|x| < k$ o $k < |x|$, luego tiene que ser $|x| = k$.

Ahora podemos expresar formalmente que dos conjuntos son equipotentes si y sólo si tienen el mismo número de elementos:

$$x \sim y \leftrightarrow |x| = |y|.$$

A partir de aquí ya es pura rutina comprobar los hechos siguientes:

1. $x \cap y = \varnothing \to |x \cup y| = |x| + |y|$.
2. $|x \times y| = |x| \cdot |y|$.
3. $|y^x| = |y|^{|x|}$.
4. $|\mathcal{P}x| = 2^{|x|}$.

El primer teorema expresa la interpretación natural de la suma de números naturales: la suma de m y n es el número de elementos que tiene un conjunto que resulta de unir conjuntos disjuntos de cardinales m y n. Vamos a ver la prueba como ilustración:

Razonamos por inducción completa sobre y. Suponemos que el resultado es cierto para todo $y' < y$. Si $y = 0$, es decir, si $y = \varnothing$, tenemos simplemente

$$|x \cup y| = |x| = |x| + 0 = |x| + |y|.$$

Si $y \neq 0$, sea $v \in y$ y sea $y' = y \setminus \{v\}$, de modo que $y' \subset y$, $y' \neq y$, luego $y' < y$, y $x \cap y' = \varnothing$. Por hipótesis de inducción tenemos que $|x \cup y'| = |x| + |y'|$.

Notemos que una biyección[3] $f : I_{|y'|} \longrightarrow y'$ se extiende a una biyección

$$f' : I_{|y'|+1} \longrightarrow y$$

sin más que definir $f'(|y'|) = v$, lo que prueba que $|y| = |y'| + 1$. Por otra parte, tenemos que $x \cup y = (x \cup y') \cup \{v\}$ y $v \notin x \cup y'$, y el mismo argumento implica que $|x \cup y| = |x| + |y'| + 1 = |x| + |y|$. ∎

El teorema anterior se refina fácilmente al siguiente:

$$|x \cup y| = |x| + |y| \dot{-} |x \cap y|.$$

Basta tener en cuenta que podemos expresar cualquier unión como unión disjunta así:

$$x \cup y = x \cup (y \setminus x),$$

con lo que $|x \cup y| = |x| + |y \setminus x|$ y, como $y = (y \setminus x) \cup (x \cap y)$ es también una unión disjunta, resulta que $|y| = |y \setminus x| + |x \cap y|$, de donde se obtiene la conclusión.

Ahora podemos interpretar en términos de aplicaciones las desigualdades entre cardinales:

Teorema 2.22 *Se cumple:*

1. $|x| \leq |y| \leftrightarrow \bigvee f \in y^x \ f : x \longrightarrow y$ *inyectiva*,
2. $x \neq \varnothing \to (|x| \leq |y| \leftrightarrow \bigvee f \in x^y \ f : y \longrightarrow x$ *suprayectiva*).

Demostración: 1) Si $|x| \leq |y|$, es obvio que, a partir de dos biyecciones $f : I_{|x|} \longrightarrow x$, $g : I_{|y|} \longrightarrow y$, podemos construir $f^{-1} \circ g : x \longrightarrow y$ inyectiva.

Si $f : x \longrightarrow y$ inyectiva, entonces $y = f[x] \cup (y \setminus f[x])$, luego $|y| = |x| + |y \setminus f[x]|$, luego $|x| \leq |y|$.

[3]Técnicamente, cuando sabemos que existe un conjunto f que cumple algo y tomamos uno en concreto, hay que entender que lo estamos definiendo, por ejemplo, tomando el mínimo f que cumple lo requerido así:

$$f \equiv \mu f \in y'^{I_{|y'|}} \ f : I_{|y'|} \longrightarrow y' \text{ biyectiva.}$$

2) Si $f : y \longrightarrow x$ suprayectiva, podemos definir

$$g \equiv \{\langle u, v\rangle \in x \times y \mid v = \mu w \in y\ f(w) = u\},$$

y es fácil ver que $g : x \longrightarrow y$ inyectiva, luego $|x| \leq |y|$.

Si $|x| \leq |y|$, basta encontrar $f : I_{|y|} \longrightarrow I_{|x|}$ suprayectiva, pues componiéndola con biyecciones obtendremos la aplicación requerida. Tenemos que $|x| > 0$, luego $0 \in I_{|x|}$ y basta definir

$$f(u) = \begin{cases} u & \text{si } u < |x|, \\ 0 & \text{en otro caso.} \end{cases}$$

■

A partir de aquí el lector debería convencerse de que cualquier hecho razonable relativo a aplicaciones y cardinales de conjuntos finitos es demostrable en ARP.

Aplicaciones y sucesiones Ahora tenemos dos formas distintas de expresar una sucesión $x_0, \ldots, x_{n-1}$, bien como una sucesión s tal que $\ell(s) = n$, bien como una aplicación $f : I_n \longrightarrow x$, donde x es un conjunto que contiene a $\{x_0, \ldots, x_n\}$. El conjunto x^{I_n} contiene a todas las aplicaciones $f : I_n \longrightarrow x$, y vamos a ver ahora que también podemos definir un conjunto que contenga a todas las sucesiones de longitud n cuyos términos estén en un conjunto x.

Para ello consideramos el funtor definido por

$$F(f, 0) = 0, \qquad F(f, i+1) = F(f, i)^\frown f(i).$$

Una simple inducción prueba que

$$f \in x^{I_n} \to \ell(F(f, m)) = m \wedge \bigwedge i < m\ F(f, m)_i = f(i).$$

Esto nos permite definir

$$S_n(x) = \{F(f, n) \mid f \in x^{I_n}\},$$

de modo que

$$s \in S_n(x) \leftrightarrow \ell(s) = n \wedge \bigwedge i < n\ s_i \in x.$$

En efecto, si $\ell(s) = n \wedge \bigwedge i < n\ s_i \in x$, podemos definir

$$f \equiv \{z \in I_n \times x \mid \bigvee i < n\, z = \langle i, s_i\rangle\},$$

de modo que $f \in x^{I_n}$ y $\bigwedge i < n\ f(i) = s_i$, y así $s = F(f, n) \in S_n(x)$. El recíproco es más sencillo.

Más aún, tenemos una aplicación $H(x, n) : x^{I_n} \longrightarrow S_n(x)$ biyectiva dada por $h(f) = F(f, n)$.

Esto nos permite intercambiar aplicaciones por sucesiones siempre que sea necesario.

Si F es un funtor de rango $n+1$, podemos definir otro funtor del mismo rango mediante

$$F|_x(x_1,\dots,x_n) = \{\langle u, F(x_1,\dots,x_n,u)\rangle \mid u \in x\},$$

de modo que

$$F|_x(x_1,\dots,x_n) : x \longrightarrow \mathcal{R}F|_x \wedge \bigwedge u \in x\, F|_x(x_1,\dots,x_n)(u) = F(x_1,\dots,x_n,u).$$

Vamos a usar esto para formular un nuevo teorema de recursión. Definimos

$$\begin{aligned}\mathrm{Sub}(s) = \{t \in \bigcup_{i<\ell(s)} S_i(\{s_j \mid j<\ell(s)\}) \mid \bigvee k<\ell(s)\bigvee r<\ell(s)(k+r\le\ell(s)\\ \wedge \ell(t) = r \wedge \bigwedge j<r\ t_j = s_{k+j})\}.\end{aligned}$$

Así $\mathrm{Sub}(s)$ está formado por las subsucesiones t de s, en el sentido de que $s = x^\frown t^\frown y$, para ciertos x, y.

Teorema 2.23 *Si G es un funtor de rango $n+2$, existe un funtor F de rango $n+1$ tal que*

$$F(x_1,\dots,x_n,s) = G(x_1,\dots,x_n,s,F|_{\mathrm{Sub}(s)}(x_1,\dots,x_n)).$$

En otros términos: podemos definir un funtor especificando el valor de $F(x_1,\dots,x_n,s)$ supuesto definido $F(x_1,\dots,x_n,t)$ para todo $t \in \mathrm{Sub}(s)$.

Demostración: Sea $L(s) = \sum_{i<\ell(s)} (s_i+1)$. Así es claro que

$$t \in \mathrm{Sub}(s) \to L(t) < L(s).$$

Sea

$$S_k = \{s \in \bigcup_{i\le k} S_i(I_k) \mid L(s) \le k\},$$

de modo que $s \in S_k \leftrightarrow L(s) \le k$. Notemos que

$$L(s) = k+1 \to \mathrm{Sub}(s) \subset S_k.$$

Definimos un funtor H de rango $n+1$ tal que

$$H_k(x_1,\dots,x_n) : S_k \longrightarrow \mathcal{R}H_k(x_1,\dots,x_n).$$

Como $S_0 = \{0\}$, podemos definir

$$H_0(x_1,\dots,x_n) = \{\langle 0, G(x_1,\dots,x_n,0,0)\rangle\}.$$

Supuesto definido $H_k(x_1,\dots,x_n)$, tomamos $s \in S_{k+1}$ y distinguimos dos casos: si $L(s) \le k$, entonces $s \in S_k$ y podemos definir

$$H_{k+1}(x_1,\dots,x_n)(s) = H_k(x_1,\dots,x_n)(s).$$

Esto garantiza que

$$H_k(x_1,\dots,x_n) \subset H_{k+1}(x_1,\dots,x_n)$$

de donde, a su vez,

$$k < r \to H_k(x_1,\dots,x_n) \subset H_r(x_1,\dots,x_n).$$

Si $L(s) = k+1$, tenemos que $\mathrm{Sub}(s) \subset S_k$, luego podemos considerar el conjunto

$$A \equiv \{\langle u, H_k(x_1,\dots,x_n)(u))\rangle \mid u \in \mathrm{Sub}(s)\},$$

y definimos $H_{k+1}(x_1,\dots,x_n)(s) = G(x_1,\dots,x_n,s,A)$.

Esto completa la definición del funtor H, y a su vez podemos definir

$$F(x_1,\dots,x_n,s) = H_{L(s)}(x_1,\dots,x_n)(s),$$

y es fácil ver que F cumple lo requerido (pues en el caso $k = 0$ se cumple que $F|_{\mathrm{Sub}(0)}(x_1,\dots,x_n) = 0$ y en el caso $k > 0$ el conjunto A que hemos considerado no es sino $F|_{\mathrm{Sub}(s)}(x_1,\dots,x_n)$. ∎

Observemos además que si dos funtores F_1 y F_2 satisfacen el teorema anterior, entonces podemos probar que

$$F_1(x_1,\dots,x_n,s) = F_2(x_1,\dots,x_n,s).$$

En efecto, si suponemos que $F_1(x_1,\dots,x_n,s) \neq F_2(x_1,\dots,x_n,s)$, podemos considerar

$$n_0 \equiv \mu m \leq L(s) \bigvee t \in S_m \; F_1(x_1,\dots,x_n,t) \neq F_2(x_1,\dots,x_n,t)$$

A su vez, podemos tomar

$$t_0 \equiv \mu t \in S_{m_0} \; F_1(x_1,\dots,x_n,t) \neq F_2(x_1,\dots,x_n,t),$$

de modo que $F_1(x_1,\dots,x_n,t_0) \neq F_2(x_1,\dots,x_n,t_0)$, pero si $L(u) < L(t_0) = m_0$, entonces $F_1(x_1,\dots,x_n,u) = F_2(x_1,\dots,x_n,u)$. Como

$$u \in \mathrm{Sub}(t_0) \to L(u) < L(t_0),$$

resulta que $F_1|_{\mathrm{Sub}(t_0)}(x_1,\dots,x_n) = F_2|_{\mathrm{Sub}(t_0)}(x_1,\dots,x_n)$, y entonces la hipótesis sobre F_1 y F_2 nos da que $F_1(x_1,\dots,x_n,t_0) = F_2(x_1,\dots x_n,t_0)$, con lo que tenemos una contradicción. ∎

Sumas finitas (bis) En este apartado trabajaremos bajo las premisas que describimos a continuación. Supondremos que hemos fijado una cierta fórmula $\phi(x,x_1,\dots,x_n)$ que abreviaremos $x \in A$ (sin mencionar explícitamente los parámetros $x_1,\dots,x_n$) y que tenemos términos

$$0 \equiv 0(x_1,\dots,x_n), \qquad x+y \equiv +(x,y,x_1,\dots,x_n)$$

de modo que las premisas serán:

1. $+ : A \times A \longrightarrow A \equiv u \in A \wedge v \in A \to u + v \in A$,
2. $u \in A \wedge v \in A \wedge w \in A \to (u + v) + w = u + (v + w)$,
3. $0 \in A$,
4. $u \in A \to u + 0 = u$.
5. $u \in A \wedge v \in A \to u + v = v + u$.

Para cada término t, definimos:

$$\sum_{i<0} t(i) = 0, \qquad \sum_{i<n+1} t(i) = \sum_{i<n} t(i) + t(n).$$

Es fácil ver (por inducción sobre n) que

$$\bigwedge i < n\ t(i) \in A \to \sum_{i<n} t(i) \in A,$$

$$\bigwedge i < n\ (t(i) \in A \wedge t(i) = t'(i)) \to \sum_{i<n} t(i) = \sum_{i<n} t'(i).$$

$$\bigwedge i < m + n\ t(i) \in A \to \sum_{i<m+n} t(i) = \sum_{i<m} t(i) + \sum_{i<n} t(m + i).$$

Conviene observar que estos hechos no requieren la premisa 5. Ésta hace falta, en cambio, para probar que

$$\bigwedge f \in I_n^{I_n}(f : I_n \longrightarrow I_n \text{ biyectiva} \wedge \bigwedge i < n\ t(i) \in A \to \sum_{i<n} t(f(i)) = \sum_{i<n} t(i)).$$

Razonamos por inducción sobre n. Para $n = 0$ es trivial, pues la igualdad que hay que probar se reduce a $0 = 0$. Supongamos que la propiedad es cierta para n y que

$$f : I_{n+1} \longrightarrow I_{n+1} \text{ biyectiva} \wedge \bigwedge i < n + 1\ t(i) \in A.$$

Sea $k \equiv f^{-1}(n) < n + 1$, de modo que $k \leq n$ y $f(k) = n$. Sea $r = n \dot{-} k$, de modo que $n + 1 = k + 1 + r$. Así:

$$\sum_{i<n+1} t(f(i)) = \sum_{i<k+1} t(f(i)) + \sum_{i<r} t(f(i)) = \sum_{i<k} t(f(i)) + t(n) + \sum_{i<r} t(f(k+1+i)).$$

Ahora definimos $g : I_n \longrightarrow I_n$ biyectiva mediante

$$g(i) = \begin{cases} f(i) & \text{si } i < k, \\ f(i+1) & \text{si } k \leq i. \end{cases}$$

Entonces

$$\sum_{i<n+1} t(f(i)) = \sum_{i<k} t(g(i)) + \sum_{i<r} t(g(k+i)) + t(n) = \sum_{i<n} t(g(i)) + t(n),$$

donde hemos usado que, como se prueba sin dificultad,

$$\bigwedge i < n\ (t(i) \in A \wedge t(i) = s(i)) \to \sum_{i<n} t(i) = \sum_{i<n} s(i).$$

Ahora aplicamos a g la hipótesis de inducción, de modo que

$$\sum_{i<n+1} t(f(i)) = \sum_{i<n} t(i) + t(n) = \sum_{i<n+1} t(i).$$

Esto nos permite definir sumas finitas en las que los índices de los sumandos varían en un conjunto arbitrario. Concretamente, llamando

$$B(x) = \mu f \in x^{I_{|x|}}\ f : |x| \longrightarrow x \text{ biyectiva},$$

definimos

$$\sum_{u\in x} t(u) = \sum_{i<|x|} t(B(x)(i)).$$

Así,

$$\bigwedge u \in x\ t(u) \in A \wedge f : I_n \longrightarrow x \text{ biyectiva} \to \sum_{u\in x} t(u) = \sum_{i<n} t(f(i)).$$

En efecto, si suponemos

$$\bigwedge u \in x\ t(u) \in A \wedge f : I_n \longrightarrow x \text{ biyectiva},$$

llamamos $g \equiv B(x)$, de modo que también $g : I_n \longrightarrow x$ biyectiva. Por lo tanto, por el teorema precedente,

$$\sum_{u\in x} t(u) = \sum_{i<n} t(g(i)) = \sum_{i<n} t(g((f \circ g^{-1})(i)) = \sum_{i<n} t(f(i)).$$

En particular

$$\sum_{u\in\varnothing} t(u) = 0, \qquad \sum_{u\in\{v\}} t(u) = t(v).$$

Con esto ya es fácil probar:

$$x \cap y = \varnothing \wedge \bigwedge u \in x \cup y\ \ t(u) \in A \to \sum_{u\in x\cup y} t(u) = \sum_{u\in x} t(u) + \sum_{u\in y} t(u).$$

En efecto, sean $m \equiv |x|$, $n \equiv |y|$, $f \equiv B(x)$, $g \equiv B(y)$, de modo que

$$f : I_m \longrightarrow x \text{ biyectiva}, \qquad g : I_n \longrightarrow y \text{ biyectiva}.$$

Sean

$$g^* \equiv \{(n+i, g(i)) \mid i \in I_n\}, \qquad h \equiv f \cup g^*,$$

de modo que es fácil ver que $h : I_{m+n} \longrightarrow x \cup y$ biyectiva. Por el resultado precedente, tenemos que

$$\sum_{u\in x\cup y} t(u) = \sum_{i<m+n} t(h(i)) = \sum_{i<m} t(h(i)) + \sum_{i<n} t(h(m+i)) =$$

$$\sum_{i<m+n} t(h(i)) = \sum_{i<m} t(f(i)) + \sum_{i<n} t(g(i)) = \sum_{u\in x} t(u) + \sum_{u\in y} t(u).$$

Con estos resultados ya podemos probar de forma natural cualquier resultado elemental relacionado con sumas finitas.

2.3 Números enteros

En principio hemos diseñado ARP para hablar de números naturales, pero nada nos impide hablar en ella de números enteros o racionales. Veamos en esta sección cómo podemos introducir los números enteros:

Definición 2.24 Consideramos la relación dada por:

$$(a,b)\,R\,(c,d) \equiv a+d = b+c.$$

A partir de las propiedades de la suma es fácil probar que

$$(a,b)\,R\,(a,b), \qquad (a,b)\,R\,(c,d) \to (c,d)\,R\,(a,b),$$

$$(a,b)\,R\,(c,d) \wedge (c,d)\,R\,(e,f) \to (a,b)\,R\,(e,f).$$

La primera propiedad nos permite definir

$$[a,b] \equiv \mu x \leq \langle a,b\rangle \bigvee uv < x(x = \langle u,v\rangle \wedge (u,v)\,R\,(a,b)).$$

Como $x = \langle a,b\rangle$ cumple la propiedad requerida, el mínimo así definido también la cumple y, de hecho,

$$[a,b] = [c,d] \leftrightarrow (a,b)\,R\,(c,d).$$

En efecto, si $[a,b] = [c,d]$, llamamos $u = [a,b]_0$, $v = [a,b]_1$, de modo que $[a,b] = \langle u,v\rangle$ y $(u,v)\,R\,(a,b)$. Pero también $(u,v)\,R\,(c,d)$, luego $(a,b)\,R\,(c,d)$.

Recíprocamente, si $(a,b)\,R\,(c,d)$, tenemos que

$$(u,v)\,R\,(a,b) \leftrightarrow (u,v)\,R\,(c,d),$$

Si aplicamos esto a $u = [a,b]_0$, $v = [a,b]_1$, $u' = [c,d]_0$, $v' = [c,d]_1$, tenemos que $(u',v')\,R\,(c,d)$, luego $(u',v')\,R\,(a,b)$, luego $[a,b] \leq [c,d]$, e igualmente se prueba la desigualdad contraria. ∎

Definición 2.25 Llamaremos *números enteros* a los conjuntos (números naturales) de la forma $[a,b]$. Explícitamente:[4]

$$z \in \mathbb{Z} \equiv z = [z_0, z_1].$$

Definimos:

$$z + z' \equiv [z_0 + z'_0, z_1 + z'_1], \qquad zz' \equiv [z_0z'_0 + z_1z'_1, z_0z'_1 + z_1z'_0],$$

$$z \leq z' \equiv z_0 + z'_1 \leq z_1 + z'_0.$$

[4]Notemos que el $\in$ que aparece en $z \in \mathbb{Z}$ no es la pertenencia entre conjuntos que habíamos definido, sino que $z \in \mathbb{Z}$ debe entenderse como la fórmula de ARP con z como única variable indicada en el miembro derecho.

Se cumple entonces que

$$[a,b]+[c,d]=[a+c,b+d], \qquad [a,b]\cdot[c,d]=[ac+bd,ad+bc],$$

$$[a,b]\le[c,d] \leftrightarrow a+d\le b+c.$$

En efecto, llamemos $z=[a,b]$, $z'=[c,d]$. Entonces

$$(z_0,z_1)\,R\,(a,b), \qquad (z'_0,z'_1)\,R\,(c,d),$$

luego $z_0+b=z_1+a$, $z'_0+d=z'_1+c$, de donde

$$z_0+z'_0+b+d=z_1+z'_1+a+c,$$

luego $(z_0+z'_0,z_1+z'_1)\,R\,(a+c,b+d)$, luego

$$[a,b]+[c,d]=[z_0+z'_0,z_1+z'_1]=[a+c,b+d].$$

Con el producto y la relación de orden se razona análogamente.

Definimos

$$0\equiv[0,0], \qquad 1\equiv[1,0], \qquad -z\equiv[z_1,z_0].$$

Nota Notemos que los números enteros 0 y 1 que acabamos de definir no coinciden con los números naturales 0 y 1, pero en lo sucesivo el contexto dejará siempre claro si por $0,1$ nos referimos a los números naturales $0,1$ o a los números enteros $0,1$, igual que podemos distinguir si hablamos de la suma, el producto y el orden de los números naturales o de los enteros. ∎

Ahora podemos probar:

Teorema 2.26 *Si $x,y,z\in\mathbb{Z}$, se cumple:*

1. $(x+y)+z=x+(y+z)$,
2. $x+y=y+x$,
3. $x+0=x$,
4. $x+(-x)=0$,
5. $(xy)z=x(yz)$,
6. $xy=yx$,
7. $x\cdot 1=x$,
8. $x(y+z)=xy+xz$,
9. $x\le x$,
10. $x\le y\wedge y\le x\to x=y$,

11. $x \leq y \wedge y \leq z \to x \leq z$,

12. $x \leq y \vee y \leq x$,

13. $x \leq y \to x + z \leq y + z$,

14. $x \geq 0 \wedge y \geq 0 \to xy \geq 0$.

Todas las comprobaciones son rutinarias. Veamos la primera igualdad, por ejemplo:

$$(x + y) + z = [x_0 + y_0, x_1 + y_1] + [z_0, z_1] = [x_0 + y_0 + z_0, x_1 + y_1 + z_1]$$

y $x + (y + z)$ lleva a la misma expresión. ∎

A cada número entero $z \in \mathbb{Z}$ le podemos asignar un *valor absoluto* como sigue:

$$|z| = \begin{cases} z_0 \dot{-} z_1 & \text{si } z_0 \geq z_1, \\ z_1 \dot{-} z_0 & \text{si } z_0 \leq z_1. \end{cases}$$

Por otro lado, definimos $+n \equiv [n, 0]$, de modo que $+0$ y $+1$ coinciden con los números enteros que hemos llamado 0 y 1, respectivamente. Escribiremos $-n \equiv -(+n)$. En general, tenemos:

Teorema 2.27 *Se cumple:*

1. $+n \in \mathbb{Z}$,

2. $+m = +n \leftrightarrow m = n$,

3. $+(m + n) = +m + (+n)$,

4. $+(mn) = (+m)(+n)$,

5. $m \leq n \leftrightarrow +m \leq +n$,

6. $z \in \mathbb{Z} \wedge z \geq 0 \to z = +|z|$,

7. $z \in \mathbb{Z} \wedge z \leq 0 \to z = -|z|$.

Demostración: 1) Es inmediato. Para probar 2) observamos que

$$+m = +n \leftrightarrow [m, 0] = [n, 0] \leftrightarrow m + 0 = n + 0 \leftrightarrow m = n.$$

3) $+m + (+n) = [m, 0] + [n, 0] = [m + n, 0 + 0] = +(m + n)$. La prueba de 4) y 5) es similar.

Para probar 6) observamos que $z \geq 0$ equivale a $z_0 \geq z_1$, con lo que

$$z_0 = z_1 + (z_0 \dot{-} z_1) = z_1 + |z|,$$

y esto equivale a $z = [z_0, z_1] = [|z|, 0] = +|z|$. El razonamiento para 7) es análogo. ∎

Observemos ahora que $y \leq z \leftrightarrow -z \leq -y$, pues

$$y < z \leftrightarrow y + (-y) + (-z) < z + (-z) + (-y) \leftrightarrow 0 - z < 0 - y \leftrightarrow -z < -y.$$

Con todo esto ya podemos afirmar que los números enteros son:

$$\cdots \quad < -3 < -2 < -1 < 0 < +1 < +2 < +3 < \quad \cdots$$

y en la práctica escribiremos n en vez de $+n$, con lo que los números enteros pueden expresarse así:

$$\cdots \quad < -3 < -2 < -1 < 0 < 1 < 2 < 3 < \quad \cdots$$

Por lo tanto, si llamamos *números enteros positivos* (resp. *negativos*) a los que cumplen

$$z \in \mathbb{Z}^+ \equiv z \in \mathbb{Z} \wedge z > 0, \qquad z \in \mathbb{Z}^- \equiv z \in \mathbb{Z} \wedge z < 0,$$

resulta que

$$z \in \mathbb{Z} \to z \in \mathbb{Z}^- \vee z = 0 \vee z \in \mathbb{Z}^+$$

y no pueden darse a la vez dos de estos tres casos, y podemos identificar los números naturales con el 0 y los números enteros positivos, en el sentido de que cualquier suma como $2+3=5$, o cualquier producto como $2 \cdot 3 = 6$, o cualquier desigualdad como $2 \leq 5$ es correcta o no tanto si consideramos que los números son naturales o enteros no negativos.

Más precisamente, si estamos considerando números naturales y necesitamos verlos como enteros, sólo tenemos que cambiar cada n por $+n$ y, si estamos considerando números enteros no negativos y necesitamos verlos como naturales, sólo tenemos que cambiar cada z por $|z|$.

Ahora podemos caracterizar el valor absoluto de un número entero como

$$|z| = \begin{cases} z & \text{si } z \geq 0, \\ -z & \text{si } z \leq 0, \end{cases}$$

y es fácil probar:

1. $|z| \geq 0, \qquad |z| = 0 \leftrightarrow z = 0,$
2. $|w + z| \leq |w| + |z|,$
3. $|wz| = |w||z|.$

Estas propiedades nos dan una prueba rápida de otro hecho relevante de la aritmética de los números enteros, y es que

$$wz = 0 \leftrightarrow w = 0 \vee z = 0.$$

A partir de los resultados que hemos enunciado aquí se pueden probar todos los hechos básicos sobre la aritmética de los números enteros de forma natural, es decir, sin necesidad de tecnicismos propios de ARP (más allá de la necesidad de que todas las pruebas de existencia sean constructivas).

Por ejemplo, para probar que es posible dividir euclídeamente números enteros definimos

$$c^*(D,d) = \begin{cases} c(|D|,|d|) & \text{si } D \geq 0,\ d > 0 \text{ o } D < 0,\ d < 0,\ r(|D|,|d|) = 0, \\ -c(|D|,|d|) & \text{si } D \geq 0,\ d < 0 \text{ o } D < 0,\ d > 0,\ r(|D|,|d|) = 0, \\ -c(|D|,|d|) - 1 & \text{si } D < 0,\ d > 0,\ r(|D|,|d|) > 0, \\ c(|D|,|d|) + 1 & \text{si } D < 0,\ d < 0,\ r(|D|,|d|) > 0, \end{cases}$$

$$r^*(D,d) = \begin{cases} r(|D|,|d|) & \text{si } D \geq 0 \text{ o } r(|D|,|d|) = 0, \\ d - r(|D|,|d|) & \text{si } D < 0,\ d > 0,\ r(|D|,|d|) > 0, \\ -d - r(|D|,|d|) & \text{si } D < 0,\ d < 0,\ r(|D|,|d|) > 0, \end{cases}$$

donde c y r son los funtores que dan el cociente y el resto de la división euclídea de números naturales. A partir de aquí escribiremos c y r en lugar de c^* y r^*. Entonces

Teorema 2.28 *Se cumple:*

$$D, d \in \mathbb{Z} \wedge d \neq 0 \to D = dc(D,d) + r(D,d) \wedge 0 \leq r(D,d) < |d|,$$

$$D, d, c, r \in \mathbb{Z} \wedge d \neq 0 \wedge D = dc + r \wedge 0 \leq r < |d| \to c = c(D,d) \wedge r = r(D,d).$$

Demostración: Sólo hay que distinguir los cuatro casos posibles para el signo de D y d. Por ejemplo, si $D < 0$, $d > 0$, tenemos que

$$|D| = |d|c(|D|,|d|) + r(|D|,|d|) \wedge 0 \leq r(|D|,|d|) < |d|,$$

luego, multiplicando por -1,

$$D = d(-c(|D|,|d|)) - r(|D|,|d|)$$

Si $r(|D|,|d|) = 0$ esto ya es $D = dc(D,d) + r(D,d)$. Si $r(|D|,|d|) > 0$, entonces

$$D = d(-c(|D|,|d|) - 1) + d - r(|D|,|d|) = dc(D,d) + r(D,d),$$

y además $0 < r(|D|,|d|) < d$, luego $0 \leq d - r(|D|,|d|) = r(D,d) < d = |d|$. Los otros casos se tratan análogamente.

Similarmente, si se cumple $D = dc + r \wedge 0 \leq r < |d|$, en el caso $D < 0$, $d > 0$, o bien $r = 0$, en cuyo caso $|D| = |d|(-c) + 0$, con lo que, por 2.2, tiene que ser $c = -c(|D|,|d|) = c(D,d)$ y $r = r(|D|,|d|) = r(D,d)$, o bien $0 < r < d$, en cuyo caso

$$|D| = |d|(-c) - r = |d|(-c-1) + d - r, \qquad 0 \leq d - r < d,$$

con lo que nuevamente $c(|D|,|d|) = -c - 1$, $r(|D|,|d|) = d - r$, de donde se concluye igualmente que $c = c(D,d)$, $r = r(D,d)$. ∎

Intervalos Terminamos con un tecnicismo conjuntista: Si n es un número natural, tenemos definido el conjunto I_n de los números naturales menores que n. El teorema 2.17 nos permite definir los conjuntos

$$I_n^* = \{+u \mid u \in I_n\}, \qquad -I_n^* = \{-u \mid u \in I_n\},$$

de donde a su vez podemos formar el conjunto $]-n, n[= I_n^* \cup I_n^*$, y es fácil ver que

$$x \in]-n, n[\leftrightarrow x \in \mathbb{Z} \wedge -n < x < n \leftrightarrow x \in \mathbb{Z} \wedge |x| < n.$$

Más en general, podemos definir

$$]m, n[\equiv \{z \in]-\max\{|m|+1, |n|+1\}, \max\{|m|+1, |n|+1\}[\mid m < z \wedge z < n\},$$

de modo que

$$m \in \mathbb{Z} \wedge n \in \mathbb{Z} \to (z \in]m, n[\leftrightarrow (z \in \mathbb{Z} \wedge m < z < n)).$$

A su vez, podemos definir

$$[m, n] \equiv]m, n[\cup \{m, n\}, \quad]m, n] \equiv]m, n[\cup \{n\}, \quad [m, n[\equiv]m, n[\cup \{m\},$$

de modo que cumplen las variantes obvias del teorema precedente.

Si definimos

$$\leq_x \equiv \{\langle u, v\rangle \in x \times x \mid u \in \mathbb{Z} \wedge v \in \mathbb{Z} \wedge u \leq v\}$$

y $x \subset \mathbb{Z} \equiv \bigwedge u \in x\, u \in \mathbb{Z}$, tenemos claramente que

$$x \subset \mathbb{Z} \to \text{c.t.o.}(x, \leq_x),$$

en el sentido definido en la sección anterior, lo que nos permite hablar del mínimo y el máximo de cualquier conjunto de números enteros.

2.4 Números racionales

Veamos ahora que también es posible hablar de números racionales en la Aritmética Recursiva Primitiva. La construcción es muy similar a la que acabamos de ver para los números enteros:

Definición 2.29 Consideramos las relaciones dadas por

$$P(a, b) \equiv a \in \mathbb{Z} \wedge b \in \mathbb{Z} \wedge b \neq 0,$$

$$(a, b)\, R\, (c, d) \equiv P(a, b) \wedge P(c, d) \wedge ad = bc.$$

A partir de las propiedades del producto es fácil probar que

$$P(a,b) \to (a,b)\,R\,(a,b), \qquad (a,b)\,R\,(c,d) \to (c,d)\,R\,(a,b),$$

$$(a,b)\,R\,(c,d) \wedge (c,d)\,R\,(e,f) \to (a,b)\,R\,(e,f).$$

Definimos la *fracción* de *numerador* a y *denominador* b como

$$a/b \equiv \mu x \leq \langle a,b\rangle + \langle -a,-b\rangle \ \bigvee uv < x(x = \langle u,v\rangle \wedge (u,v)\,R\,(a,b) \wedge v > 0).$$

Así, supuesto $P(a,b)$, tenemos que $(a,b)\,R\,(a,b)$ y $(-a,-b)\,R\,(a,b)$ y, o bien $b > 0$, o bien $-b > 0$, luego o bien $x = \langle a,b\rangle$ o bien $x = \langle -a,-b\rangle$ cumple la propiedad requerida, luego el mínimo que define la fracción también la cumple y, de hecho,

$$P(a,b) \wedge P(c,d) \to \left(\frac{a}{b} = \frac{c}{d} \leftrightarrow ad = bc\right).$$

Llamaremos *números racionales* a los conjuntos (números naturales) de la forma a/b, con $a,b \in \mathbb{Z}$, $b \neq 0$. Explícitamente:

$$q \in \mathbb{Q} \equiv \bigvee ab < q\ (a \in \mathbb{Z} \wedge b \in \mathbb{Z} \wedge b \neq 0 \wedge q = a/b).$$

Más precisamente, si $q \in \mathbb{Q}$, por definición tenemos que $q = q_0/q_1$ con $q_1 > 0$.

Definimos

$$q + r \equiv \frac{q_0 r_1 + q_1 r_2}{q_1 r_1}, \qquad qr \equiv \frac{q_0 r_0}{q_1 r_1},$$

$$q \leq r \equiv q_0 r_1 \leq q_1 r_0.$$

Se comprueba fácilmente que si $P(a,b)$ y $P(c,d)$, entonces

$$\frac{a}{b} + \frac{c}{d} = \frac{ad + bc}{bd}, \qquad \frac{a}{b}\frac{c}{d} = \frac{ac}{bd},$$

$$b > 0 \wedge d > 0 \to \left(\frac{a}{b} \leq \frac{c}{d} \leftrightarrow ad \leq bc\right).$$

Definimos:

$$0 \equiv \frac{0}{1}, \qquad 1 \equiv \frac{1}{1}, \qquad -q \equiv \frac{-q_0}{q_1}, \qquad q^{-1} \equiv \frac{q_1}{q_0}.$$

Así, si $a,b \in \mathbb{Z}$, se cumple que

$$-\frac{a}{b} = \frac{-a}{b}, \qquad b \neq 0 \to \left(\frac{a}{b}\right)^{-1} = \frac{b}{a}.$$

Es fácil probar las propiedades siguientes:

Teorema 2.30 *Si $q,r,s \in \mathbb{Q}$, se cumple:*

1. $(q + r) + s = q + (r + s)$,
2. $q + r = r + q$,

3. $q + 0 = q$,
4. $q + (-q) = 0$,
5. $(qr)s = q(rs)$,
6. $qr = rq$,
7. $q \cdot 1 = q$,
8. $q(r + s) = qr + qs$,
9. $q \neq 0 \to qq^{-1} = 1$,
10. $q \leq q$,
11. $q \leq r \wedge r \leq q \to q = r$,
12. $q \leq r \wedge r \leq s \to q \leq s$,
13. $q \leq r \vee r \leq q$,
14. $q \leq r \to q + s \leq r + s$,
15. $q \geq 0 \wedge r \geq 0 \to qr \geq 0$.

Si $x, y \in \mathbb{Z}$, tenemos que

$$\frac{x}{1} = \frac{y}{1} \leftrightarrow x = y, \qquad \frac{x}{1} + \frac{y}{1} = \frac{x+y}{1}, \qquad \frac{x}{1} \cdot \frac{y}{1} = \frac{xy}{1}, \qquad \frac{x}{1} \leq \frac{y}{1} \leftrightarrow x \leq y.$$

Esto nos permite identificar los números enteros con los números racionales de la forma $x/1$. Técnicamente, definimos

$$x \in \tilde{\mathbb{Z}} \equiv \bigvee m < x(m \in \mathbb{Z} \wedge x = m/1), \qquad +\tilde{n} \equiv \frac{+n}{1},$$

y a partir de ahora llamaremos $\mathbb{Z}$ y $+n$ a lo que acabamos de definir como $\tilde{\mathbb{Z}}$ y $+\tilde{n}$, respectivamente.

A partir de estas propiedades se pueden probar fácilmente todas las propiedades básicas de la aritmética de los números racionales.

2.5 Números primos

Como último ejemplo de las posibilidades de ARP para formalizar la aritmética elemental vamos a demostrar los resultados básicos sobre números primos. En primer lugar definimos la relación de divisibilidad:

Definición 2.31 $x \mid y \equiv \bigvee u \leq y \; y = xu$.

Es fácil probar las propiedades elementales siguientes:

Teorema 2.32 *Se cumple:*

1. $1 \mid x \wedge x \mid x \wedge x \mid 0$,
2. $x \mid y \wedge y \neq 0 \to x \leq y$,
3. $x \mid y \wedge y \mid z \to x \mid z$,
4. $x \mid y \wedge y \mid x \to x = y$,
5. $x \mid y \to x \mid yz$,
6. $x \mid y \wedge x \mid z \to x \mid (y + z)$.

DEMOSTRACIÓN: 1) es inmediato. Para probar 2) observamos que si $y = xz$ con $y \neq 0$, entonces $z \neq 0$, luego $1 \leq z$, luego $x \leq xz = y$.

Para 4) observamos que si $y = ux \wedge x = vy$ entonces, descartando el caso trivial en que $x = y = 0$, tenemos que $y = uvy$, luego $uv = 1$ y esto implica $u = 1$ por 2). Todo lo demás es sencillo. ∎

Ahora podemos definir los números primos:

Definición 2.33 $\text{primo}(p) \equiv p > 1 \wedge \bigwedge uv \leq p(p = uv \to u = 1 \vee v = 1)$.

Claramente:

$$\text{primo}(p) \wedge x \mid p \to x = 1 \vee x = p.$$

Veamos ahora que existen primos. Más aún, todo número $x > 1$ tiene al menos un divisor primo:

Teorema 2.34 $x > 1 \to \bigvee p \leq x(p \mid x \wedge \text{primo}(p))$.

DEMOSTRACIÓN: Si $x > 1$, tenemos que el conjunto

$$\{y \leq x \mid y > 1 \wedge y \mid x\}$$

no es vacío, pues contiene a x, luego podemos tomar

$$p = \text{mín}\{y \leq x \mid y > 1 \wedge y \mid x\},$$

que es el menor divisor no trivial de x. En particular, $p \leq x$. Sólo tenemos que probar que es primo. Ciertamente, $p > 1$ y, si $p = uv$, pero $u \neq 1$ y $v \neq 1$, entonces $u \mid x$, $u \neq 0$, $u \neq 1$ y $u < p$, en contra de la minimalidad de p. ∎

Dos números son *coprimos* si cumplen:

$$\text{cop}(x, y) \equiv \bigwedge p \leq x(\text{primo}(p) \wedge p \mid x \to p \nmid y).$$

Equivalentemente, dos números son coprimos si no tienen factores primos comunes. El resultado fundamental es:

Teorema 2.35 $n \mid ab \wedge \text{cop}(n, a) \to n \mid b$.

DEMOSTRACIÓN: Vamos a probar:

$$\bigwedge abn \leq m(m = ab \wedge n \mid m \wedge \mathrm{cop}(n, a) \to n \mid b)$$

por inducción sobre m. Suponemos, pues, que el resultado es cierto cuando $m < ab$, y vamos a probarlo para ab. Tenemos que $n \mid ab$, luego existe un q tal que $nq = ab$. Distinguimos tres casos:

Si $n = a$, entonces $\mathrm{cop}(n, n)$ implica que $n = 1$ (pues de lo contrario sabemos que existe un primo que divide a n), y entonces obviamente $n \mid b$.

Si $n < a$, tenemos que $nq = ab = nb + (a \dot{-} n)b$, de donde $nb < nq$, luego $b < q$. Por lo tanto,

$$nb + n(q \dot{-} b) = nq = nb + (a \dot{-} n)b,$$

con lo que $n(q \dot{-} b) = (a \dot{-} n)b$ y así $n \mid (a \dot{-} n)b < ab$. Además, $\mathrm{cop}(n, a \dot{-} n)$, pues si un primo cumpliera $p \mid n$ y $p \mid a \dot{-} n$, también cumpliría $p \mid a$, en contra de que n y a son coprimos. Por hipótesis de inducción, concluimos que $n \mid b$.

Por último, si $n > a$, tenemos que $nq = aq + (n \dot{-} a)q = ab$, luego $aq < ab$, luego $q < b$. Por lo tanto,

$$aq + (n \dot{-} a)q = ab = aq + a(b \dot{-} q),$$

con lo que $(n \dot{-} a)q = a(b \dot{-} q)$ y así $n \dot{-} a \mid a(b \dot{-} q) < ab$. Como en el caso anterior, $n \dot{-} a$ y a son coprimos, luego por hipótesis de inducción $n \dot{-} a \mid b \dot{-} q$. Pongamos que $b \dot{-} q = (n \dot{-} a)d$, con lo que $(n \dot{-} a)q = a(n \dot{-} a)d$, luego $q = ad$, luego $ab = nq = nad$, luego $b = nd$, y así tenemos también que $n \mid b$. ∎

De aquí se deduce la propiedad fundamental de los números primos:

Teorema 2.36 $\mathrm{primo}(p) \wedge p \mid ab \to p \mid a \vee p \mid b$.

DEMOSTRACIÓN: Basta observar que si $p \nmid a$, entonces $\mathrm{cop}(p, a)$, pues si un primo cumple $q \mid p$ y $q \mid a$, necesariamente $q = p$ y tenemos que $p \mid a$. Por lo tanto, podemos aplicar el teorema anterior. ∎

Para probar que hay infinitos primos usamos el argumento de Euclides, que requiere definir previamente el funtor *factorial* como

$$0! = 1, \qquad (n + 1)! = (n + 1) \cdot n!$$

Una simple inducción prueba que $\bigwedge i \leq n\ i \mid n!$, lo que a su vez implica que

$$\bigwedge i \leq n\, i \nmid n! + 1,$$

luego $\mu p(n! + 1) > n$. Esto permite definir el *siguiente primo* como

$$\mathrm{sp}(n) = \mu p \leq \mu p(n! + 1)\ (\mathrm{primo}(p) \wedge n < p),$$

y a su vez el funtor dado por

$$p_0 = 2, \quad p_{n+1} = \mathrm{sp}(p_n),$$

que recorre los números primos. En efecto, se cumple:

$$\mathrm{primo}(p_n),$$

$$\mathrm{primo}(p) \to \bigvee n < p\, p = p_n.$$

La primera fórmula se prueba fácilmente por inducción, y para la segunda se prueba primero que $n < p_n$, con lo que, dado un primo $p > 0$, se cumple $p < p_p$, luego podemos tomar el mínimo m tal que $p < p_m$, y no puede ser $m = 0$, con lo que $n = m \dot{-} 1$ cumple $p_n \leq p < p_{n+1} = \mathrm{sp}(p_n)$, y esto implica que $p = p_n$.

Si p es primo, en particular $p \geq 2$, luego $a < p^a$, luego $p^a \nmid a$. Esto nos permite definir

$$e(p, a) = \mu n \leq a \mid p^n \nmid a,$$

y no puede ser $e(p, a) = 0$, luego podemos definir

$$v_p(a) = \mu n \leq a \mid p^n \nmid a \dot{-} 1,$$

con lo que

$$\mathrm{primo}(p) \to p^{v_p(a)} \mid a \wedge p^{v_p(a)+1} \nmid a.$$

Es fácil ver entonces que

$$\mathrm{primo}(p) \to (p^n \mid a \leftrightarrow n \leq v_p(a)),$$

de modo que $v_p(a)$ es el máximo exponente con el que p divide a a. Se cumple:

1. $\mathrm{primo}(p) \to v_p(ab) = v_p(a) + v_p(b)$,
2. $\mathrm{primo}(p) \to v_p(a + b) \geq \mathrm{mín}\{v_p(a), v_p(b)\}$.

En efecto, tenemos que $a = p^{v_p(a)}c$, $b = p^{v_p(b)}c'$, donde $c \equiv c(a, p^{v_p(a)})$, $c' \equiv c(b, p^{v_p(b)})$, luego $ab = p^{v_p(a)+v_p(b)}cc'$ y esto implica que

$$v_p(ab) \geq v_p(a) + v_p(b).$$

Por otro lado,

$$ab = p^{v_p(ab)}c'' = p^{v_p(a)+v_p(b)}cc'.$$

Si fuera $v_p(ab) > v_p(a) + v_p(b)$ deduciríamos que $p \mid cc'$, luego $p \mid c$ o $p \mid c'$. Si, por ejemplo, se da el primer caso, de ahí se sigue que $p^{v_p(a)+1} \mid a$, lo cual es imposible.

La segunda parte se prueba análogamente a como hemos obtenido la primera desigualdad en la primera parte.

Factorizaciones en primos En el último apartado de la sección 2.2 definimos las sumas finitas

$$\sum_{i\in x} t(i)$$

a partir de una fórmula $x \in A$ y de unos términos 0, $x + y$ que cumplieran ciertas premisas. Consideramos la definición correspondiente a $x \in A \equiv x > 0$ y a los términos 1 y $x \cdot y$, que satisfacen todas las premisas requeridas. En este contexto las "sumas finitas" que hemos definidos las llamaremos productos finitos, y usaremos la notación

$$\prod_{i\in x} m_i,$$

donde $m_i \equiv t(i)$ es un término arbitrario. No se trata de que aquí tengamos que dar una definición análoga, sino que meramente estamos cambiando la notación que usamos en la sección 2.2. Con esta notación, los resultados que hemos probado se expresan así:

$$\bigwedge i \in x\, m_i > 0 \to \prod_{i\in x} m_i > 0,$$

$$\prod_{i\in\varnothing} m_i = 1, \qquad m_j > 0 \to \prod_{i\in\{j\}} m_i = m_j,$$

$$x \cap y = \varnothing \wedge \bigwedge i \in x \cup y\; m_i > 0 \to \prod_{i\in x\cup y} m_i = \prod_{i\in x} m_i \cdot \prod_{i\in y} m_i.$$

Si llamamos $r(a)$ al máximo número natural tal que $p_{r(a)} \mid a$, ahora podemos probar el teorema siguiente:

$$a > 1 \to a = \prod_{n\le r(a)} p_n^{v_{p_n}(a)}.$$

Razonamos por inducción sobre a. Si $a \le 2$ no hay nada que probar. Si es cierto para números menores que a, tomamos el mínimo n_0 tal que $p_{n_0} \mid a$, de modo que $a = p_{n_0}^{v_{p_{n_0}}(a)} a'$, y claramente $v_{p_{n_0}}(a') = 0$. En cambio, si $q \ne p_{n_0}$ es cualquier otro primo, tiene que ser $v_q(p_{n_0}^{v_{p_{n_0}}(a)}) = 0$, luego $v_q(a') = v_q(a)$. Aplicando la hipótesis de inducción, es fácil concluir que

$$a = p_{n_0}^{v_{p_{n_0}}(a)} \cdot \prod_{n\le r(a')} p_n^{v_{p_n}(a')} = \prod_{n\le r(a)} p_n^{v_{p_n}(a)}.$$

Así pues, todo número natural se descompone en producto de factores primos. En particular, esto implica que

$$\bigwedge n \le \max\{r(a), r(b)\}(v_{p_n}(a) = v_{p_n}(b)) \to a = b.$$

Para probar la unicidad de la descomposición conviene expresar el teorema de factorización en otros términos. Para ello definimos

$$F(a) \equiv \{f \in \mathcal{P}(I_{a+1} \times I_{a+1}) \mid \bigvee n \le a+1\; f : I_n \longrightarrow I_{a+1}\}$$

y así podemos acotar el cuantificador del teorema siguiente:

$$a > 1 \to \bigvee p \in F(a) \bigvee n \leq a+1 (p : I_n \longrightarrow I_{a+1} \wedge$$
$$\bigwedge i < n \ \text{primo}(p_i) \wedge a = \prod_{i \in I_n} p_i).$$

La prueba, por inducción sobre a, es similar a la anterior, pero ahora podemos enunciar así la unicidad:

$$\bigwedge i \in x \ \text{primo}(p_i) \wedge \bigwedge j \in y \ \text{primo}(q_i) \wedge \prod_{i \in x} p_i = \prod_{j \in y} q_j \to$$
$$\bigvee f \in y^x (f : x \longrightarrow y \ \text{biyectiva} \wedge \bigwedge i \in x \ p_i = q_{f(i)})).$$

Para probar esto, primeramente probamos este otro hecho:

$$\text{primo}(p) \wedge \bigwedge i \in x \, m_i > 0 \wedge p \mid \prod_{i \in x} m_i \to \bigvee i \in x \ p \mid m_i.$$

Esto se prueba por inducción sobre $|x'|$ para $x' \in \mathcal{P}x$. Si $|x'| = 0$ es trivial porque no puede darse la hipótesis. En caso contrario tomamos $i_0 \equiv \mu i \ i \in x'$. Si $x' = \{i_0\}$ la conclusión es trivial, pues entonces $p \mid m_{i_0}$. En caso contrario podemos expresar

$$\prod_{i \in x'} m_i = m_{i_0} \cdot \prod_{i \in x' \setminus \{i_0\}} m_i$$

y el hecho de que p sea primo implica que $p \mid m_{i_0}$ o bien $p \mid \prod_{i \in x' \setminus \{i_0\}} m_i$.

En el primer caso ya tenemos la conclusión, y en el segundo basta aplicar la hipótesis de inducción.

Con esto, la unicidad de las descomposiciones en primos se prueba también por inducción sobre $|x'|$. Si $|x'| = 0$ los productos valen 1 y es fácil ver entonces que $|y| = 0$ o, de lo contrario, un primo dividiría a 1. La conclusión se cumple trivialmente en este caso.

Supongamos que es cierto cuando $|x'| = n$ y supongamos que $|x'| = n + 1$. Tomamos $i_0 \in x'$, de modo que

$$p_{i_0} \cdot \prod_{i \in x' \setminus \{i_0\}} p_i = \prod_{j \in y} q_j,$$

luego por el resultado previo, existe un $j_0 \in y$ tal que $p_{i_0} \mid q_{j_0}$, pero, como ambos son primos, esto implica que $p_{i_0} = q_{j_0}$. De aquí se sigue que

$$\prod_{i \in x' \setminus \{i_0\}} p_i = \prod_{j \in y \setminus \{j_0\}} q_j,$$

luego, por hipótesis de inducción, existe $f : x' \setminus \{i_0\} \longrightarrow y \setminus \{j_0\}$ biyectiva tal que $\bigwedge i \in x' \setminus \{i_0\} \ p_i = q_{f(i)}$ y basta extender f a una biyección $f : x' \longrightarrow y$ mediante $f(i_0) = j_0$. Claramente se cumple la conclusión.

Capítulo III

La formalización de la lógica

En el capítulo anterior hemos visto cómo en ARP se puede formalizar la aritmética y la teoría de conjuntos finitos. Ahora vamos a ver que también podemos formalizar en ella la propia construcción de ARP (pero no los conceptos semánticos, es decir, los relacionados con los números denotados por términos y la satisfacción de fórmulas). Esto significa que toda la construcción informal de ARP no presupone ningún principio más allá de lo que expresan los axiomas y reglas de inferencia de ARP (La formalización de la semántica de ARP la estudiaremos más adelante.) Seguidamente definiremos en ARP la lógica de primer orden, es decir, una clase muy general de teorías axiomáticas capaces de formalizar prácticamente cualquier contexto matemático.

3.1 La formalización de ARP en ARP

En la definición 1.1 hemos tomado ocho signos arbitrarios como signos del lenguaje $\mathcal{L}_{\text{arp}}$. A la hora de formalizar esta definición en ARP podemos tomar como signos ocho números naturales arbitrarios, pero por simplicidad fijaremos ocho en concreto:

Definición 3.1 Llamaremos *signos* de $\ulcorner\mathcal{L}_{\text{arp}}\urcorner$ a

$$\ulcorner 0\urcorner \equiv 0, \quad \ulcorner S\urcorner \equiv 1, \quad \ulcorner p\urcorner \equiv 2, \quad \ulcorner c\urcorner \equiv 3,$$

$$\ulcorner \kappa\urcorner \equiv 4, \quad \ulcorner \rho\urcorner \equiv 5, \quad \ulcorner x\urcorner \equiv 6, \quad \ulcorner =\urcorner \equiv 7.$$

El conjunto de signos de $\ulcorner\mathcal{L}_{\text{arp}}\urcorner$ es, pues, el conjunto

$$\text{Sig}(\ulcorner\mathcal{L}_{\text{arp}}\urcorner) \equiv \{\ulcorner 0\urcorner, \quad \ulcorner S\urcorner, \quad \ulcorner p\urcorner, \quad \ulcorner c\urcorner, \quad \ulcorner \kappa\urcorner, \quad \ulcorner \rho\urcorner, \quad \ulcorner x\urcorner, \quad \ulcorner =\urcorner\}.$$

Usamos ángulos de Quine $\ulcorner\ \urcorner$ para distinguir los signos metamatemáticos de sus formalizaciones. Así, S es un funtor del lenguaje de ARP, mientras que $\ulcorner S\urcorner \equiv S0$ es un numeral.

Cadenas de signos Diremos que ζ es una *cadena de signos de* $\ulcorner\mathcal{L}_{\rm arp}\urcorner$ si cumple

$$\zeta \in \mathrm{Cad}(\ulcorner\mathcal{L}_{\rm arp}\urcorner) \equiv \bigwedge i < \ell(\zeta)\ \zeta_i \in \mathrm{Sig}(\ulcorner\mathcal{L}_{\rm arp}\urcorner).$$

Esta definición incluye a la cadena de signos vacía 0 de longitud 0.

Claramente, la *yuxtaposición* $\zeta_1{}^\frown\zeta_2$ de dos cadenas de signos es también una cadena de signos que representaremos simplemente como $\zeta_1 \cdot \zeta_2$ o incluso $\zeta_1\zeta_2$ cuando no haya confusión posible.

Es fácil ver que la fórmula $x \in \mathrm{Cad}(\ulcorner\mathcal{L}_{\rm arp}\urcorner)$ junto con los términos 0 (la sucesión vacía) y $\zeta_1\zeta_2$ cumplen las premisas enunciadas en la página 98 para definir las sumas finitas (sin contar la última, la propiedad conmutativa, que no era necesaria para probar las propiedades más básicas de las sumas finitas), que aquí representaremos con notación multiplicativa, de modo que tenemos definido el término

$$\prod_{i<n} \zeta_i$$

(donde $\zeta_i \equiv \zeta(i, x_1, \ldots, x_n)$ es, en principio, un término cualquiera) y se cumple

$$\bigwedge i < n\ \zeta_i \in \mathrm{Cad}(\ulcorner\mathcal{L}_{\rm arp}\urcorner) \to \prod_{i<n} \zeta_i \in \mathrm{Cad}(\ulcorner\mathcal{L}_{\rm arp}\urcorner),$$

$$\bigwedge i < n\ (t(i) \in \mathrm{Cad}(\ulcorner\mathcal{L}_{\rm arp}\urcorner) \wedge \zeta_i = \zeta_i') \to \prod_{i<n} \zeta_i = \prod_{i<n} \zeta_i'.$$

$$\bigwedge i < m+n\ \zeta_i \in \mathrm{Cad}(\ulcorner\mathcal{L}_{\rm arp}\urcorner) \to \prod_{i<m+n} \zeta_i = \prod_{i<m} \zeta_i \prod_{i<n} \zeta_{m+i}.$$

De la propia definición se sigue además que

$$\zeta_0 \in \mathrm{Cad}(\ulcorner\mathcal{L}_{\rm arp}\urcorner) \to \prod_{i<1} \zeta_i = \zeta_0.$$

En la práctica escribiremos

$$\zeta_0 \cdots \zeta_{n-1} \equiv \prod_{i<n} \zeta_i$$

o, mejor aún, para adaptarnos a la costumbre de empezar a contar en 1 en lugar de en 0, escribiremos $\zeta_1 \cdots \zeta_n$ en lugar de $\zeta_0 \cdots \zeta_{n-1}$. Esto sólo es un convenio "estético" que en cualquier momento podemos eliminar sin alterar el contenido matemático de cualquier afirmación que hagamos.

Si $\zeta \equiv s_1 \cdots s_n$ es cualquier cadena metamatemática de signos s_i de $\mathcal{L}_{\rm arp}$, llamaremos $\ulcorner\zeta\urcorner$ al único numeral que según el teorema 1.16, cumple

$$\underset{\rm ARP}{\vdash} \ulcorner\zeta\urcorner = \ulcorner s_1\urcorner \cdots \ulcorner s_n\urcorner = \langle \ulcorner s_1\urcorner, \ldots, \ulcorner s_n\urcorner \rangle\,.$$

Nota Con esto hemos introducido una ambigüedad en la notación. En el caso de un signo de $\mathcal{L}_{\text{arp}}$, como por ejemplo p, la notación $\ulcorner p \urcorner$ puede hacer referencia a dos numerales distintos. Según la definición 3.1 es $\ulcorner p \urcorner = 2$, pero según la definición que acabamos de dar (viendo a p como una cadena de signos de longitud 1) tenemos que $\langle \ulcorner p \urcorner \rangle = \langle 2 \rangle = 4$, luego $\ulcorner p \urcorner \equiv 4$. En general, según este segundo criterio, tenemos que

$$\ulcorner 0 \urcorner \equiv 1, \quad \ulcorner S \urcorner \equiv 2, \quad \ulcorner p \urcorner \equiv 4, \quad \ulcorner c \urcorner \equiv 7,$$

$$\ulcorner \kappa \urcorner \equiv 11, \quad \ulcorner \rho \urcorner \equiv 16, \quad \ulcorner x \urcorner \equiv 22, \quad \ulcorner = \urcorner \equiv 29.$$

En la práctica el contexto siempre dejará claro cuándo consideramos que una expresión como $\ulcorner s \urcorner$ hace referencia a un signo de $\ulcorner \mathcal{L}_{\text{arp}} \urcorner$ o a la cadena de signos de longitud 1 correspondiente. ∎

Numerales Diremos que N es un *numeral* si cumple

$$N \in \text{Num}(\ulcorner \mathcal{L}_{\text{arp}} \urcorner) \equiv \ell(N) > 0 \wedge \bigwedge i < \ell(N) \dot{-} 1 \, N_i = \ulcorner S \urcorner \wedge N_{\ell(N) \dot{-} 1} = \ulcorner 0 \urcorner.$$

Consideramos el funtor definido por

$$N_0 = \ulcorner 0 \urcorner, \qquad N_{n+1} = \ulcorner S \urcorner N_n.$$

Una simple inducción prueba que $N_n \in \text{Num}(\ulcorner \mathcal{L}_{\text{arp}} \urcorner)$ y, más aún, los numerales de esta forma son los únicos que hay, pues si definimos el *indice* de un numeral como $\text{ind}(N) = \ell(N) \dot{-} 1$, tenemos que $\text{ind}(N_n) = n$ y que

$$N \in \text{Num}(\ulcorner \mathcal{L}_{\text{arp}} \urcorner) \wedge \text{ind}(N) = n \to N = N_n.$$

Es claro que si $\bar{n}$ es el numeral (metamatemático) correspondiente al número natural n, entonces en ARP se demuestra que $\ulcorner \bar{n} \urcorner = N_{\bar{n}}$. Por ejemplo: $\ulcorner SSSS0 \urcorner \equiv N_4$. Esto se prueba por inducción (metamatemática) sobre n.

Notemos que no es lo mismo el numeral (metamatemático) $4 \equiv SSSS0$ que el numeral (formalizado) $N_4 = \ulcorner SSSS0 \urcorner \equiv 352\,942\,601$.

Por otro lado, el funtor N que hemos definido un poco más arriba es algo más general que la sucesión de numerales $\ulcorner 0 \urcorner, \ulcorner S0 \urcorner, \ulcorner SS0 \urcorner, \ldots$, pues esto son únicamente términos sin variables, mientras que el funtor N nos permite considerar el término N_n, donde n es una variable de ARP, una variable que al sustituirla por un numeral, como $\bar{4} \equiv SSSS0$, da lugar a un término (que se demuestra que es) igual al numeral $\ulcorner SSSS0 \urcorner$.

Proyecciones Definimos el funtor diádico

$$p^n_k = \ulcorner p \urcorner N_k N_n.$$

Llamaremos *proyecciones* a las cadenas de signos que cumplen

$$\pi \in \text{Proy}(\ulcorner \mathcal{L}_{\text{arp}} \urcorner) \equiv \pi \in \text{Cad}(\ulcorner \mathcal{L}_{\text{arp}} \urcorner) \wedge \bigvee kn < \ell(\pi)(1 \le k \le n \wedge \pi = p^n_k).$$

Si $\pi \in \mathrm{Proy}(\ulcorner \mathcal{L}_{\mathrm{arp}} \urcorner)$, podemos calcular:

$$k(\pi) = (\mu i < \ell(\pi)\ \pi_i = \ulcorner 0 \urcorner) \dot{-} 1, \qquad n(\pi) = \ell(\pi) \dot{-} (k(\pi) + 3),$$

y es fácil ver que

$$\pi \in \mathrm{Proy}(\ulcorner \mathcal{L}_{\mathrm{arp}} \urcorner) \to \pi = p^{n(\pi)}_{k(\pi)}.$$

Los números $k(\pi)$ y $n(\pi)$ se llaman *índice* y *rango* de π, respectivamente. Por definición tenemos que $1 \leq k(\pi) \leq n(\pi)$.

Es fácil ver que si $1 \leq k \leq n$ son números naturales metamatemáticos, entonces

$$\underset{\mathrm{ARP}}{\vdash} \ulcorner p^n_k \urcorner = p^{\bar{n}}_{\bar{k}},$$

donde $\bar{k}$ y $\bar{n}$ son los numerales (metamatemáticos) de ARP correspondientes a k y n, respectivamente.

Funtores Vamos a definir un funtor monádico rang usando el teorema 2.23, es decir, que definiremos $\mathrm{rang}(\zeta)$ suponiendo definido el rango de las cadenas de $\mathrm{Sub}(\zeta)$. Definimos $\mathrm{rang}(0) = 0$ y, para cadenas de longitud no nula, distinguimos varios casos:

1. $\mathrm{rang}(\ulcorner S \urcorner) = 1$, $\mathrm{rang}(\ulcorner c \urcorner) = 1$.
2. Si $\zeta_0 = \ulcorner p \urcorner$ y existen números $1 \leq k \leq \ell(\zeta)$ tales que $\zeta = p^n_k$, entonces $\mathrm{rang}(\zeta) = n$.
3. Si $\zeta_0 = \ulcorner \kappa \urcorner$ y existen $1 \leq m, n \leq \ell(\zeta)$ y $h \in \mathrm{Sub}(\zeta)$ tales que $\mathrm{rang}(h) = m$ y existe $g \in S_m(\mathrm{Sub}(\zeta))$ de modo que

   $$\bigwedge i < m\ \mathrm{rang}(g_i) = n, \qquad \zeta = \ulcorner \kappa \urcorner h g_1 \cdots g_m,$$

   entonces $\mathrm{rang}(\zeta) = n$.
4. Si $\zeta_0 = \ulcorner \rho \urcorner$ y existe $1 \leq n \leq \ell(\zeta)$ y cadenas $g, h \in \mathrm{Sub}(\zeta)$ de modo que $\mathrm{rang}(g) = n$, $\mathrm{rang}(h) = n + 2$ y $\zeta = \ulcorner \rho \urcorner g h$, entonces $\mathrm{rang}(\zeta) = n + 1$.

En cualquier otro caso, $\mathrm{rang}(\zeta) = 0$. Llamaremos *funtores* de $\ulcorner \mathcal{L}_{\mathrm{arp}} \urcorner$ a las cadenas de signos que cumplen

$$f \in \mathrm{Fun}(\ulcorner \mathcal{L}_{\mathrm{arp}} \urcorner) \equiv f \in \mathrm{Cad}(\ulcorner \mathcal{L}_{\mathrm{arp}} \urcorner) \wedge \mathrm{rang}(f) > 0.$$

Así, cada funtor $f \in \mathrm{Fun}(\ulcorner \mathcal{L}_{\mathrm{arp}} \urcorner)$ tiene asociado un rango $\mathrm{rang}(f) > 0$.

A partir de las definiciones precedentes (y suprimiendo precisiones técnicas), es fácil ver que todo funtor $f \in \mathrm{Fun}(\ulcorner \mathcal{L}_{\mathrm{arp}} \urcorner)$ se encuentra en uno de los casos siguientes:

1. $f = \ulcorner S \urcorner$ (en cuyo caso f es un funtor monádico),
2. $f = \ulcorner c \urcorner$ (en cuyo caso f es un funtor monádico),

3. $f = p^n_k$, para ciertos índices $1 \leq k \leq n$ (y entonces f es un funtor n-ádico),
4. $f = \ulcorner\kappa\urcorner(h, g_1, \ldots, g_m)$, donde h es un funtor m-ádico y los g_i forman una sucesión de funtores n-ádicos (y entonces f es un funtor n-ádico),
5. $f = \ulcorner\rho\urcorner(g, h)$, donde g es un funtor n-ádico y h es un funtor $n+2$-ádico (y entonces f es un funtor $n+1$-ádico).

Es evidente que un mismo funtor no puede estar a la vez en en dos de estos casos, pues el signo inicial f_0 es distinto en cada uno de ellos, pero, más aún, en los casos 4 y 5 los funtores que componen f también están unívocamente determinados.

En efecto, en primer lugar definimos el funtor

$$\mathrm{R}(\zeta, n) = \mu\zeta' \in \mathrm{Sub}(\zeta)\bigvee\zeta'' \in \mathrm{Sub}(\zeta)(\ell(\zeta'') = n \wedge \zeta = \zeta''\zeta').$$

Es fácil ver que si $\zeta \in \mathrm{Cad}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner)$ y $\ell(\zeta) \geq n$, entonces $R(\zeta, n)$ es la cadena que resulta de quitarle a ζ sus n primeros signos.

A su vez definimos

$$\mathrm{PN}(\zeta) = \mu\zeta' \in \mathrm{Sub}(\zeta)\bigvee k \leq \ell(\zeta)(\zeta' = N_k \wedge \zeta = N_k\ R(\zeta, k+1)).$$

Así, $\mathrm{PN}(\zeta)$ es el numeral con el que empieza ζ si es que realmente empieza por un numeral, y es 0 (que es también la cadena vacía) si no es el caso. Es fácil probar entonces que

$$\zeta \in \mathrm{Cad}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner) \to \mathrm{PN}(N_k\zeta) = N_k.$$

Esto implica que el numeral inicial de una cadena de signos (si existe) está unívocamente determinado, es decir, que una misma cadena de signos no puede descomponerse de dos formas distintas de modo que el primer fragmento sea un numeral.

Ahora definimos un funtor monádico PF con la propiedad de que, si f es un funtor, entonces

$$\mathrm{PF}(f\zeta) = f,$$

es decir, que PF proporciona el primer funtor de su argumento, mientras que si ζ es una cadena que no empieza por un funtor, entonces $\mathrm{PF}(\zeta) = 0$ es la cadena vacía.

Para ello definimos $\mathrm{PF}(0) = 0$ y, para cadenas de longitud no nula, distinguimos varios casos:

1. Si $\zeta_0 = \ulcorner S\urcorner$, entonces $\mathrm{PF}(\zeta) = \ulcorner S\urcorner$,
2. Si $\zeta_0 = \ulcorner c\urcorner$, entonces $\mathrm{PF}(\zeta) = \ulcorner c\urcorner$,
3. Si $\zeta_0 = \ulcorner p\urcorner$, calculamos $\bar{k} \equiv \mathrm{PN}(R(\zeta, 1))$. Si no es la cadena vacía, calculamos $\bar{n} \equiv \mathrm{PN}(R(\zeta, 1 + \ell(\bar{k})))$. Si no es la cadena vacía y además $2 \leq \ell(\bar{k}) \leq \ell(\bar{n})$,
$$\mathrm{PF}(\zeta) = \zeta|_{\ell(\bar{k})+\ell(\bar{n})+1}.$$

4. Si $\zeta_0 = \ulcorner\kappa\urcorner$, calculamos $h \equiv \mathrm{PF}(R(\zeta,1))$. Si no es un funtor, entonces $\mathrm{PF}(\zeta) = 0$. En caso contrario, llamamos $m \equiv \mathrm{rang}(h)$ y consideramos el funtor definido por $F(\zeta,0) = \langle \ulcorner\kappa\urcorner h, R(\zeta, 1+\ell(h))\rangle$ y

   $$F(\zeta,n+1) = \langle F(\zeta,n)_0 \frown \mathrm{PF}(F(\zeta,n)_1), R(F(\zeta,n)_1, \ell(\mathrm{PF}(F(\zeta,n)_1)))\rangle$$
   si $\mathrm{PF}(F(\zeta,n)_1) \neq 0$, o $F(\zeta,n+1) = \langle 0,0\rangle$ si $\mathrm{PF}(F(\zeta,n)_1) = 0$.

   Así, si $F(\zeta,m) = \langle 0,0\rangle$, definimos $\mathrm{PF}(\zeta) = 0$ y en caso contrario hacemos $\mathrm{PF}(\zeta) = F(\zeta,m)_0$.

   Lo que hace F es quitarle a ζ la cadena inicial $\ulcorner\kappa\urcorner h$ y luego ir quitándole funtores mientras sea posible. Lo que hemos hecho es decir que si se le pueden quitar m funtores, entonces $\mathrm{PF}(\zeta)$ es la yuxtaposición de $\ulcorner\kappa\urcorner$ y los $m+1$ funtores que hemos quitado.

5. Si $\zeta_0 = \ulcorner\rho\urcorner$, calculamos $g = \mathrm{PF}(R(\zeta,1))$. Si es un funtor, calculamos $h = \mathrm{PF}(R(\zeta,1+\ell(h)))$. Si es un funtor y $\mathrm{rang}(h) = \mathrm{rang}(g)+2$, definimos $\mathrm{PF}(\zeta) = \ulcorner\rho\urcorner gh$.

En cualquier otro caso, $\mathrm{PF}(\zeta) = 0$.

Ahora una inducción rutinaria muestra que cumple lo requerido, es decir, que si f es un funtor y ζ es una cadena de signos arbitraria, entonces $\mathrm{PF}(f\zeta) = f$, lo que prueba que el primer funtor de una cadena de signos está unívocamente determinado.

Usando esto es inmediato comprobar que la descomposición de un funtor como composición o recursión es única.

Observemos también que una simple inducción sobre la longitud de f prueba que si f es un funtor (metamatemático) de rango n, entonces

$$\underset{\mathrm{ARP}}{\vdash} \ulcorner f\urcorner \in \mathrm{Fun}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner) \wedge \mathrm{ran}(\ulcorner f\urcorner) = \bar{n},$$

donde $\bar{n}$ es el numeral (metamatemático) asociado a n.

Enumeración de funtores Es interesante observar que podemos enumerar los funtores de $\ulcorner\mathrm{ARP}\urcorner$. Para ello observamos que si $f \in \mathrm{Fun}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner)$, entonces $\ulcorner\kappa\urcorner(\ulcorner S\urcorner, f)$ es otro funtor del mismo rango, pero mayor como número natural (por el teorema 2.10). Esto nos permite definir los funtores

$$\mathrm{SF}(f) = \mu g \leq \ulcorner\kappa\urcorner(\ulcorner S\urcorner, f)(g \in \mathrm{Fun}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner) \wedge f < g),$$

$$\mathrm{SFR}(f) = \mu g \leq \ulcorner\kappa\urcorner(\ulcorner S\urcorner, f)(g \in \mathrm{Fun}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner) \wedge \mathrm{rang}(g) = \mathrm{rang}(f) \wedge f < g).$$

A su vez:

$$f_0 = \ulcorner S\urcorner, \qquad f_{k+1} = \mathrm{SF}(f_k),$$

$$f_0^n = \mu g \leq p_1^n(g \in \mathrm{Fun}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner) \wedge \mathrm{rang}(g) = n), \qquad f_{k+1} = \mathrm{SFR}(f_k),$$

de modo que

$$f_k \in \mathrm{Fun}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner), \qquad f_k^n \in \mathrm{Fun}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner) \wedge \mathrm{rang}(f_k^n) = n,$$

$$k < k' \to f_k < f_{k'} \wedge f^n_k < f^n_{k'},$$

(de donde se sigue por inducción que $k \leq f_k$, $k \leq f^n_k$), y a su vez

$$f \in \mathrm{Fun}(\ulcorner \mathcal{L}_{\mathrm{arp}} \urcorner) \to \overset{1}{\bigvee} k \leq f \; f = f_k,$$

$$f \in \mathrm{Fun}(\ulcorner \mathcal{L}_{\mathrm{arp}} \urcorner) \wedge \mathrm{rang}(f) = n \to \overset{1}{\bigvee} k \leq f \; f = f^n_k.$$

Veamos, como ejemplo, la prueba de la primera de estas dos propiedades. Dado un funtor f, tomando $m \equiv f$, se cumple que $f \leq f_m < f_{m+1}$, luego podemos considerar $r \equiv \mu m \leq f + 1(f < f_m)$. Claramente, no puede ser $r = 0$, luego si hacemos $k = r \dot{-} 1$, tenemos que $f_k \leq f < f_{k+1} = \mathrm{SF}(f_k)$. La definición de SF implica entonces que $f = f_k$.

Nota Estamos demostrando teoremas en ARP, pero no debemos olvidar que cada teorema de ARP tiene una interpretación natural que es, necesariamente, una afirmación verdadera sobre los números naturales y las funciones recursivas primitivas. En particular, hemos demostrado que, si identificamos cada signo de ARP con un número natural y, por consiguiente, cada funtor de ARP con una sucesión finita de números naturales, que a su vez puede identificarse con un número natural, existe una función recursiva primitiva que enumera todos los funtores de ARP y, como cada funtor denota una función recursiva primitiva (aunque funtores distintos pueden denotar la misma función), resulta que tenemos una forma explícita de enumerar (con repeticiones) todas las funciones recursivas primitivas, $F_0, F_1, F_2, F_3, \ldots$

Este es relevante porque en la introducción señalamos que, pese a la poca importancia que un formalista puede darle al uso de "para todo", lo cierto es que el hecho de que podamos hablar informalmente de conceptos como "conjunto", "función", "propiedad", "definición", etc., esto no justifica que hablemos de "todos los conjuntos", "todas las funciones", etc., ya que no tenemos ninguna forma de atribuir un significado preciso a una afirmación sobre la totalidad de los conjuntos, las funciones, etc., más allá de los casos en los que tenemos un argumento aplicable a cualquier caso concreto. La posibilidad de enumerar todas las funciones recursivas primitivas se traduce en que podemos hablar informalmente de la totalidad de ellas, puesto que afirmar que todas las funciones recursivas primitivas cumplen algo equivale a afirmar que F_0 lo cumple, y también F_1, y también F_2, etc. No podemos asegurar que vayamos a estar en condiciones de asegurar si esto sucede o no, pero lo importante es que sabemos lo que significa. ∎

Variables Consideramos el funtor definido por

$$x_n = \ulcorner x \urcorner N_n.$$

Llamaremos *variables* a las cadenas de signos que cumplen

$$\xi \in \mathrm{Var}(\ulcorner \mathcal{L}_{\mathrm{arp}} \urcorner) \equiv \xi \in \mathrm{Cad}(\ulcorner \mathcal{L}_{\mathrm{arp}} \urcorner) \wedge \bigvee n \leq \ell(\xi)\, \xi = x_n.$$

El *índice* de una variable es $\mathrm{ind}(\xi) = \ell(\xi) \dot{-} 2$. Es fácil probar que

$$\xi \in \mathrm{Var}(\ulcorner \mathcal{L}_{\mathrm{arp}} \urcorner) \wedge \mathrm{ind}(\xi) = n \to \xi = x_n.$$

Si x es la variable (metamatemática) de índice n, es fácil ver que

$$\underset{\mathrm{ARP}}{\vdash} \ulcorner x \urcorner = x_{\bar{n}},$$

donde $\bar{n}$ es el numeral (metamatemático) correspondiente a n.

Términos Para definir los términos de $\ulcorner \mathcal{L}_{\mathrm{arp}} \urcorner$ definimos un funtor F que tome el valor 1 sobre los términos y 0 en otro caso. Escribiendo

$$\zeta \in \mathrm{Term}(\ulcorner \mathcal{L}_{\mathrm{arp}} \urcorner) \equiv F(\zeta) = 1,$$

la definición es:

1. Si $\zeta \in \mathrm{Var}(\ulcorner \mathcal{L}_{\mathrm{arp}} \urcorner)$, entonces $\zeta \in \mathrm{Term}(\ulcorner \mathcal{L}_{\mathrm{arp}} \urcorner)$.
2. Si $\zeta = \ulcorner 0 \urcorner$, entonces $\zeta \in \mathrm{Term}(\ulcorner \mathcal{L}_{\mathrm{arp}} \urcorner)$.
3. Si $f \equiv \mathrm{PF}(\zeta)$ es un funtor, llamamos $m \equiv \mathrm{rang}(f)$ y establecemos que $\zeta \in \mathrm{Term}(\ulcorner \mathcal{L}_{\mathrm{arp}} \urcorner)$ si

$$\bigvee t \in S_m(\mathrm{Sub}(\zeta))(\bigwedge i < m\ t_i \in \mathrm{Term}(\ulcorner \mathcal{L}_{\mathrm{arp}} \urcorner) \wedge \zeta = f t_1 \cdots t_m).$$

En cualquier otro caso, $t \notin \mathrm{Term}(\ulcorner \mathcal{L}_{\mathrm{arp}} \urcorner)$. Es claro entonces que

$$\zeta \in \mathrm{Term}(\ulcorner \mathcal{L}_{\mathrm{arp}} \urcorner) \leftrightarrow \zeta = \ulcorner 0 \urcorner \vee \zeta \in \mathrm{Var}(\ulcorner \mathcal{L}_{\mathrm{arp}} \urcorner) \vee \bigvee f \in \mathrm{Sub}(\zeta)$$

$$\bigvee m \leq \ell(f) \bigvee t \in S_m(\mathrm{Sub}(\zeta))(f \in \mathrm{Fun}(\ulcorner \mathcal{L}_{\mathrm{arp}} \urcorner) \wedge m = \mathrm{rang}(f) \wedge$$
$$\bigwedge i < m\ t_i \in \mathrm{Term}(\ulcorner \mathcal{L}_{\mathrm{arp}} \urcorner) \wedge \zeta = f t_1 \cdots t_m).$$

Sin tecnicismos, tenemos que t es un término si y sólo si $t = \ulcorner 0 \urcorner$ o bien t es una variable, o bien existe un funtor m-ádico f y una sucesión de m términos tal que $t = f t_1 \cdots t_m$.

En la práctica, en este último caso, escribiremos $t = f(t_1, \ldots, t_m)$. Como en el caso de los funtores, es pura rutina comprobar que si t es de esta forma, entonces m y los términos t_i están unívocamente determinados. Una forma de probarlo es definir un funtor PT tal que si t es un término, entonces $\mathrm{PT}(t\zeta) = t$, es decir, que PT asigna a cada cadena el término por el que empieza, si es que existe, o 0 en caso contrario. Para ello definimos $\mathrm{PT}(0) = 0$ y, para cadenas de longitud no nula, distinguimos varios casos:

1. Si $\zeta_0 = \ulcorner 0 \urcorner$, entonces $\mathrm{PT}(\zeta) = \ulcorner 0 \urcorner$.
2. Si $\zeta_0 = \ulcorner x \urcorner$, llamamos $N \equiv \mathrm{PT}(R(\zeta, 1))$. Si $N \in \mathrm{Num}(\ulcorner \mathcal{L}_{\mathrm{arp}} \urcorner)$, entonces $\mathrm{PT}(\zeta) = \ulcorner x \urcorner N$.

3. Si $f \equiv \mathrm{PF}(\zeta)$ es un funtor, llamamos $m \equiv \mathrm{rang}(f)$ y consideramos el funtor definido por $F(\zeta, 0) = \langle f, R(\zeta, \ell(f))\rangle$ y

$$F(\zeta, n+1) = \langle F(\zeta, n)_0 \frown \mathrm{PT}(F(\zeta, n)_1), R(F(\zeta, n)_1, \ell(\mathrm{PT}(F(\zeta, n)_1)))\rangle$$

si $\mathrm{PT}(F(\zeta, n)_1) \neq 0$, o $F(\zeta, n+1) = \langle 0, 0\rangle$ si $\mathrm{PT}(F(\zeta, n)_1) = 0$.

Finalmente, si $F(\zeta, m) = \langle 0, 0\rangle$, definimos $\mathrm{PT}(\zeta) = 0$ y en caso contrario hacemos $\mathrm{PT}(\zeta) = F(\zeta, m)_0$.

Es fácil ver que el funtor PT cumple lo requerido:

$$t \in \mathrm{Term}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner) \wedge \zeta \in \mathrm{Cad}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner) \to \mathrm{PT}(t\zeta) = f,$$

es decir, que PT extrae el primer término de cualquier cadena de signos, y con eso es fácil probar la unicidad de la expresión de un término construido a partir de un funtor.

Por último observemos que una simple inducción sobre la longitud de t, prueba que si t es un término (metamatemático), entonces

$$\underset{\mathrm{ARP}}{\vdash} \ulcorner t\urcorner \in \mathrm{Term}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner).$$

Fórmulas Definimos las *fórmulas* de $\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner$ como las cadenas de signos que cumplen:

$$\alpha \in \mathrm{Form}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner) \equiv \alpha \in \mathrm{Cad}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner) \wedge \bigvee t_1 t_2 \in \mathrm{Sub}(\alpha)$$

$$(t_1, t_2 \in \mathrm{Term}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner) \wedge \alpha = \ulcorner{=}\urcorner\, t_1 t_2).$$

En la práctica escribiremos $t_1 \ulcorner{=}\urcorner\, t_2$ en lugar de $\ulcorner{=}\urcorner\, t_1 t_2$. Podemos definir los funtores

$$T_1(\alpha) = \mathrm{PT}(R(\alpha, 1)), \qquad T_2(\alpha) = R(\alpha, 1 + \ell(T_1(\alpha))),$$

de modo que

$$t_1, t_2 \in \mathrm{Term}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner) \to T_1(t_1 \ulcorner{=}\urcorner\, t_2) = t_1 \wedge T_2(t_1 \ulcorner{=}\urcorner\, t_2) = t_2.$$

$$\alpha \in \mathrm{Form}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner) \to T_1(\alpha), T_2(\alpha) \in \mathrm{Term}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner) \wedge \alpha = T_1(\alpha) \ulcorner{=}\urcorner\, T_2(\alpha).$$

Esto significa que los dos términos que componen una fórmula están unívocamente determinados.

También es claro que si α es una fórmula (metamatemática) de ARP, entonces

$$\underset{\mathrm{ARP}}{\vdash} \ulcorner\alpha\urcorner \in \mathrm{Form}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner).$$

Diremos que θ es una *expresión* de ARP si cumple

$$\theta \in \mathrm{Exp}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner) \equiv \theta \in \mathrm{Term}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner) \vee \theta \in \mathrm{Form}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner).$$

Sustitución Definimos ahora la sustitución de una variable por un término en una expresión, distinguiendo casos:

1. Si $\zeta \in \mathrm{Var}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner)$,
$$\mathtt{S}^t_x\zeta \equiv \begin{cases} t & \text{si } \zeta = x, \\ \zeta & \text{si } \zeta \neq x. \end{cases}$$

2. Si $\zeta = \ulcorner\mathtt{0}\urcorner$, entonces $\mathtt{S}^t_x\ulcorner\mathtt{0}\urcorner = \ulcorner\mathtt{0}\urcorner$.

3. Si $f = \mathrm{PF}(\zeta)$ es un funtor y $\mathrm{rang}(f) = m$ y existe $s \in S_m(\mathrm{Sub}(\zeta))$ tal que $\bigwedge i < m\ s_i \in \mathrm{Term}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner)$ y $\zeta = fs_1 \cdots s_m$, entonces definimos
$$\mathtt{S}^t_x\zeta = f\mathtt{S}^t_x s_1 \cdots \mathtt{S}^t_x s_m.$$

4. Si $\zeta_0 = \ulcorner{=}\urcorner$ y existen $t_1, t_2 \in \mathrm{Sub}(\zeta)$ tales que $\zeta = \ulcorner{=}\urcorner t_1 t_2$, definimos $\mathtt{S}^t_x\zeta = \ulcorner{=}\urcorner\mathtt{S}^t_x t_1 \mathtt{S}^t_x t_2$.

En cualquier otro caso, definimos $\mathtt{S}^t_x\zeta = 0$.
Es claro entonces que (suponiendo $x \in \mathrm{Var}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner) \wedge t \in \mathrm{Term}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner)$):

1. $\mathtt{S}^t_x\ulcorner\mathtt{0}\urcorner = \ulcorner\mathtt{0}\urcorner$.

2. $y \in \mathrm{Var}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner) \to \mathtt{S}^t_x y = \begin{cases} t & \text{si } y = x, \\ y & \text{si } y \neq x. \end{cases}$

3. $f \in \mathrm{Fun}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner) \wedge \mathrm{rang}(f) = n \wedge \ell(s) = n \wedge \bigwedge i < n\ s_i \in \mathrm{Term}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner)$
$$\to \mathtt{S}^t_x f(s_1, \ldots, s_n) = f(\mathtt{S}^t_x s_1, \ldots, \mathtt{S}^t_x s_n).$$

4. $t_1, t_2 \in \mathrm{Term}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner) \to \mathtt{S}^t_x(t_1\ulcorner{=}\urcorner t_2) = \mathtt{S}^t_x t_1 \ulcorner{=}\urcorner\ \mathtt{S}^t_x t_2$.

Es fácil ver que

$$x \in \mathrm{Var}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner) \wedge t \in \mathrm{Term}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner) \wedge \zeta \in \mathrm{Term}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner) \to \mathtt{S}^t_x\zeta \in \mathrm{Term}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner),$$

$$x \in \mathrm{Var}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner) \wedge t \in \mathrm{Term}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner) \wedge \zeta \in \mathrm{Form}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner) \to \mathtt{S}^t_x\zeta \in \mathrm{Form}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner),$$

así como que si x, t, θ son, respectivamente, una variable, un término y una expresión metamatemáticas, entonces

$$\underset{\mathrm{ARP}}{\vdash} \mathtt{S}^{\ulcorner t \urcorner}_{\ulcorner x \urcorner}\ulcorner\theta\urcorner = \ulcorner\mathtt{S}^t_x\theta\urcorner.$$

Nota: Formalización en ARP Hasta aquí hemos detallado las definiciones y las demostraciones en ARP para que el lector se convenza de que todo cuanto hemos dicho es expresable y demostrable en ARP. No obstante, no es nada práctico exponer las definiciones y demostraciones con este nivel de detalle, ni en ARP ni en cualquier otra teoría axiomática formal.

La forma habitual de trabajar de los matemáticos en cualquier teoría formal no es detallar todas las definiciones y argumentos hasta el punto de que sea evidente cómo se formalizan en una teoría axiomática en particular, sino sólo hasta el punto de que se entiendan las ideas relevantes, dando por hecho que formalizar los pasos técnicos que no ofrecen ninguna dificultad conceptual es una pura rutina que no aporta nada salvo el hecho de confirmar que lo que se está diciendo es formalizable en una teoría dada, pero normalmente uno puede convencerse de que esto es así sin necesidad de especificar todos los detalles *ad nauseam.*

Por ejemplo, en la sección 1.1 definimos la sustitución de una variable x por un término t en una expresión θ como la expresión que resulta de sustituir cada aparición de x en θ por el término t. Esto (acompañado, si acaso, de algún ejemplo) basta para que el lector entienda perfectamente qué es $\mathtt{S}^t_x\theta$ y cómo se calcula. No obstante, justo a continuación, en la sección 1.1 especificamos la forma en concreto en que se calcula $\mathtt{S}^t_x\theta$ mediante cuatro condiciones recursivas.

En esta sección también hemos enunciado esas cuatro condiciones, aunque su aspecto es aquí mucho más técnico. Por ejemplo, si comparamos la tercera:

$$f \in \mathrm{Fun}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner) \wedge \mathrm{rang}(f) = n \wedge \ell(s) = n \wedge \bigwedge i < n\ s_i \in \mathrm{Term}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner)$$
$$\to \mathtt{S}^t_x f s_1 \cdots s_n = f\mathtt{S}^t_x s_1 \cdots \mathtt{S}^t_x s_n$$

con su formulación equivalente en la sección 1.1:

$$\mathtt{S}^t_x f(t_1, \ldots, t_n) \equiv f(\mathtt{S}^t_x t_1, \ldots, \mathtt{S}^t_x t_n),$$

vemos que la diferencia consiste en que en esta versión sobreentendemos que f es un funtor n-ádico, que t es un término, etc., mientras que en la versión formalizada hemos explicitado estas hipótesis para asegurarnos de que la fórmula que presentamos es realmente un teorema de ARP, y no un mero fragmento de teorema.

En la práctica, no hay ningún inconveniente (al contrario, tiene muchas ventajas) en formular los teoremas de ARP en un lenguaje más laxo, por ejemplo diciendo (así, con palabras) que si x es una variable, t es un término, f es un funtor n-ádico y $t_1, \ldots, t_n$ son términos, entonces

$$\mathtt{S}^t_x f(t_1, \ldots, t_n) \equiv f(\mathtt{S}^t_x t_1, \ldots, \mathtt{S}^t_x t_n),$$

sin necesidad de escribir una fórmula técnica en forma de implicación que contenga todas las hipótesis, y eso es lo que haremos de ahora en adelante.

En cuanto a los argumentos, a la hora de juzgar si un determinado concepto o un determinado argumento son formalizables en ARP, el mejor método no es ponerse a escribir una formalización detallada, sino reflexionar sobre si la definición o la demostración pueden realizarse acotando siempre *a priori* los objetos que queremos considerar.

Por ejemplo, para completar la exposición sobre la sustitución nos faltaría definir la sustitución múltiple:

$$\mathtt{S}^{t_1\cdots t_n}_{x_1\cdots x_n}\theta = \mathtt{S}^{t_1}_{y_1}\cdots\mathtt{S}^{t_n}_{y_n}\mathtt{S}^{y_1}_{x_1}\cdots\mathtt{S}^{y_n}_{x_n}\theta,$$

donde $y_1,\ldots,y_n$ son variables cualesquiera que no estén en $t_1,\ldots,t_n$ ni en θ. Formalizar esta definición requiere definir funtores que calculen las variables contenidas en una sucesión de expresiones, y que calculen una biyección entre esas variables y otras nuevas, y otros que apliquen sustituciones de forma recurrente, etc., pero el lector que haya analizado las pruebas que hemos dado en esta sección debería convencerse de que no hay nada que impida detallar toda esa construcción de modo que sea obviamente formalizable en ARP, y por eso mismo sería una pérdida de tiempo ponerse a escribir todos esos detalles en un papel.

Para que el lector afine su capacidad de estimar qué es formalizable en ARP y qué no, vamos a ver a continuación un ejemplo de concepto que hemos definido en la sección 1.1, pero que no es formalizable en ARP. ∎

La (imposibilidad de) formalización de la semántica Tratemos ahora de formalizar en ARP la definición 1.3 de la función denotada por un funtor. Un posible enunciado sería el siguiente:

Teorema *Existe un funtor diádico ϕ tal que las afirmaciones siguientes son teoremas de* ARP:

1. $\phi(\ulcorner S\urcorner, s) = s_0 + 1$.
2. $\phi(\ulcorner c\urcorner, s) = 0$.
3. $1 \leq k \leq n \to \phi(p^n_k, s) = s_{k\dot{-}1}$.
4. $h \in \mathrm{Fun}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner) \wedge \mathrm{rang}(h) = m \wedge \ell(g) = m \wedge \bigwedge i < m(g_i \in \mathrm{Fun}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner) \wedge \mathrm{rang}(g_i) = n) \wedge \ell(s) = n \to \phi(\ulcorner\kappa\urcorner(h, g_1, \ldots, g_n), s) = \phi(h, \prod_{i<m}\langle\phi(g_i, s)\rangle)$.
5. $g, h \in \mathrm{Fun}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner) \wedge \mathrm{rang}(g) = n \wedge \mathrm{rang}(h) = n+2 \wedge \ell(s) = n \to$

$$\phi(\ulcorner\rho\urcorner(g,h), s^\frown\langle 0\rangle) = \phi(g,s) \wedge$$

$$\phi(\ulcorner\rho\urcorner(g,h), s^\frown\langle k+1\rangle) = \phi(h, s^\frown\langle k\rangle{}^\frown\langle\phi(\ulcorner\rho\urcorner(g,h), s^\frown\langle k\rangle)\rangle).$$

Relajando el formalismo, este "teorema" afirma que el funtor ϕ cumple (bajo las hipótesis obvias en cada caso):

1. $\phi(\ulcorner S\urcorner, \langle n\rangle) = n+1$,
2. $\phi(\ulcorner c\urcorner, \langle n\rangle) = 0$,
3. $\phi(p^n_k, \langle m_1,\ldots,m_n\rangle) = m_k$,

4. $\phi(\ulcorner\kappa\urcorner(h, g_1, \ldots, g_m), \langle s_1, \ldots, s_n\rangle) =$

$$\phi(h, \langle \phi(g_1, \langle s_1, \ldots, s_n\rangle), \ldots, \phi(g_m, \langle s_1, \ldots, s_n\rangle)\rangle),$$

5. $\phi(\ulcorner\rho\urcorner(g, h), \langle s_1, \ldots, s_n, 0\rangle) = \phi(g, \langle s_1, \ldots, s_n\rangle)$,
$\phi(\ulcorner\rho\urcorner(g, h), \langle s_1, \ldots, s_n, k+1\rangle) = \phi(h, \langle s_1, \ldots, s_n, k, \phi(\ulcorner\rho\urcorner(g, h), \langle s_1, \ldots, s_n, k\rangle)\rangle)$.

Admitiendo esto, una simple inducción sobre la longitud de f prueba que si f es un funtor n-ádico, $k_1, \ldots, k_n$ son números naturales y $\bar{k}_i$ es el numeral correspondiente a k_i, entonces

$$(*) \qquad \vdash_{\mathrm{ARP}} \phi(\ulcorner f\urcorner, \langle \bar{k}_1, \ldots, \bar{k}_n\rangle) = \overline{F(f)(k_1, \ldots, k_n)},$$

donde $F(f)$ es la función denotada por el funtor f.

Sin embargo el "teorema" anterior es falso. Vamos a demostrar que no existe un funtor ϕ que cumpla $(*)$. El lector que crea que debería poder definirse un funtor ϕ que cumpla las condiciones indicadas debería tratar de dar una definición detallada formalizable en ARP, y si la encuentra, entonces es que no ha acabado de entender qué hace falta para que una construcción sea formalizable en ARP.

Por otro lado, es fácil programar un ordenador para que si le damos como entrada un funtor n-ádico f de ARP y una n-tupla $\langle k_1, \ldots, k_n\rangle$ nos dé como respuesta $F(f)(k_1, \ldots, k_n)$. En otras palabras, la función diádica $H(f, s)$ que, cuando f, visto como sucesión de números naturales, corresponde a un funtor n-ádico de $\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner$ y s es una sucesión de longitud n, calcula

$$H(f, s) = F(f)(s_0, \ldots, s_{n-1})$$

(y para otros argumentos toma el valor 0) la puede calcular un ordenador, pero no es recursiva primitiva (si lo fuera, un funtor ϕ que la denotara cumpliría $(*)$).

Supongamos que existiera el funtor ϕ. Entonces podríamos definir un funtor monádico mediante

$$f(k) = \phi(f^1_k, \langle k\rangle) + 1.$$

En otras palabras, para calcular $f(k)$ calculamos el k-ésimo funtor monádico, lo evaluamos en k y le sumamos 1 al resultado. Tenemos entonces que

$$\ulcorner f\urcorner \in \mathrm{Fun}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner) \wedge \mathrm{rang}(\ulcorner f\urcorner) = 1.$$

Puesto que $\mu u \le \ulcorner f\urcorner(\ulcorner f\urcorner = f^1_u)$ es un término sin variables, podemos considerar el número k que denota, de modo que, según el teorema 1.16, se cumple que

$$\vdash_{\mathrm{ARP}} \bar{k} = \mu u \le \ulcorner f\urcorner(\ulcorner f\urcorner = f^1_u),$$

donde $\bar{k}$ es el numeral que denota al número k. Esto implica que

$$\vdash_{\mathrm{ARP}} \ulcorner f\urcorner = f^1_{\bar{k}}.$$

Por el mismo teorema aplicado al término $f(\bar{k})$, tenemos que

$$\underset{\text{ARP}}{\vdash} f(\bar{k}) = \overline{F(f)(k)}.$$

Combinando todo esto concluimos que en ARP se podría demostrar que

$$f(\bar{k}) = \phi(f^1_{\bar{k}}, \langle \bar{k} \rangle) + 1 = \phi(\ulcorner f \urcorner, \langle \bar{k} \rangle) + 1 = \overline{F(f)(k)} + 1 = f(\bar{k}) + 1,$$

y esto implica a su vez que $0 = 1$, lo cual no puede probarse en ARP, luego no existe el funtor ϕ. ∎

Nota En el argumento anterior subyace el hecho siguiente: si enumeramos todas las funciones monádicas recursivas primitivas, $f^1_0, f^1_1, f^1_2, \ldots$ (con posibles repeticiones), la función dada por $f(k) = f^1_k(k) + 1$ puede calcularla un ordenador, pero no es recursiva primitiva, pues entonces sería una de las f^1_k, y entonces $f^1_k(k) = f(k) = f^1_k(k) + 1$, lo cual es imposible. ∎

En particular, no es posible formalizar en ARP el concepto de "el número natural denotado por un término respecto de una valoración", es decir, no existe ningún funtor diádico tal que si $t \in \text{Term}(\ulcorner \mathcal{L}_{\text{arp}} \urcorner)$ y v es una valoración (definida, por finitud, sobre las variables que aparecen en t), entonces $\psi(t, v)$ pueda interpretarse como el número denotado por t cuando sus variables se interpretan según v.

Más precisamente, no puede suceder que ψ cumpla, en particular, que, si t es un término metamatemático sin variables, entonces

$$\underset{\text{ARP}}{\vdash} \psi(\ulcorner t \urcorner, 0) = \overline{N(t)},$$

donde $N(t)$ es el número natural denotado por t. En tal caso, el funtor

$$f(k) = \psi(f^1_k(N_k), 0) + 1,$$

donde ahora N es el funtor que a cada número natural le asigna su numeral correspondiente en $\ulcorner \text{ARP} \urcorner$, nos llevaría a la misma contradicción que antes: si $\underset{\text{ARP}}{\vdash} (\ulcorner f \urcorner = f^1_{\bar{k}})$, entonces

$$f(\bar{k}) = \psi(f^1_{\bar{k}}(N_{\bar{k}}), 0) + 1 = \psi(\ulcorner f \urcorner(\ulcorner k \urcorner), 0) + 1 = \psi(\ulcorner f(\bar{k}) \urcorner, 0) + 1$$

$$= \overline{N(f(\bar{k}))} + 1 = \overline{F(f)(k)} + 1 = f(\bar{k}) + 1.$$

A su vez, esto implica que no es posible definir un funtor diádico $\vDash \alpha[v]$ tal que si $\alpha \in \text{Form}(\ulcorner \mathcal{L}_{\text{arp}} \urcorner)$ y v es una valoración definida sobre las variables que aparecen en α, entonces $\vDash \alpha[v]$ tome el valor 1 o 0 según si α es satisfecha o no respecto de la valoración v. Y, aunque esto pudiera definirse —que no se puede— ni aun así se podría definir un funtor $\vDash \alpha$ que distinga si una fórmula es verdadera o falsa, pues tampoco podríamos formalizar en ARP el "para toda valoración" que requiere la definición.

En general, podemos decir que ARP es capaz de formalizar su propia sintaxis, pero no puede formalizar su propia semántica.

Los axiomas de $\ulcorner\mathbf{ARP}\urcorner$ Los axiomas de ARP son las definiciones de los funtores (distintos de S) presentadas en la definición 1.8. El lector debería examinarla y convencerse de que no hay ningún inconveniente que impida formalizar esa definición y convertirla en una fórmula $\alpha \in \mathrm{Ax}(\ulcorner\mathrm{ARP}\urcorner)$ de ARP. Los detalles son muy laboriosos, pero vamos a esbozarlos.

Recordemos que hemos definido x_k como la variable de índice k de $\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner$. Podemos definir la definición del funtor p_k^n como:

$$A_p(k,n) = p_k^n(x_0,\ldots,x_{n\dot{-}1}) \ulcorner{=}\urcorner\ x_{k\dot{-}1}.$$

Para dar más detalles sobre el término concreto que estamos expresando con el miembro izquierdo tendríamos que definir el funtor

$$\xi_n = \prod_{i<n} \langle x_i \rangle$$

que a cada n le asigna la sucesión de las n primeras variables de $\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner$ (más laxamente: $\xi_n = \langle x_0,\ldots,x_{n\dot{-}1}\rangle$), y definir

$$A_p(k,n) = \ulcorner{=}\urcorner\ \ulcorner p\urcorner\ N_k\, N_n\, \xi_n\, x_{k\dot{-}1},$$

pero no ganamos nada encriptando así la definición. La definición de los axiomas de $\ulcorner\mathrm{ARP}\urcorner$ empieza así:

$$\alpha \in \mathrm{Ax}(\ulcorner\mathrm{ARP}\urcorner) \equiv \alpha = \ulcorner c(x_0) = 0\urcorner \vee$$

$$\bigvee kn \leq \ell(\alpha)(1 \leq k \leq n \wedge \alpha = A_p(k,n)) \vee \cdots$$

donde falta completar la disyunción con otra fórmula correspondiente a los funtores definidos por composición y otras dos correspondientes a los funtores definidos por recursión.

Veamos cómo sería la fórmula correspondiente a las composiciones. La idea básica es que

$$A_\kappa(h,g) = \kappa(h,g_1,\ldots,g_m)(x_1,\ldots x_n) \ulcorner{=}\urcorner\ h(g_1(x_1,\ldots,x_n),\ldots,g_m(x_1,\ldots,x_n)).$$

Más técnicamente sería:

$$A_\kappa(h,g) = \ulcorner{=}\urcorner\ \ulcorner\kappa\urcorner\, h\, g\, \xi_n\, h \prod_{i<\ell(g)} g_i\xi_n.$$

y para añadir esto en la fórmula $\alpha \in \mathrm{Ax}(\ulcorner\mathrm{ARP}\urcorner)$ hay que incluir cuantificadores

$$\bigvee h \in \mathrm{Sub}(\alpha)(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner)\bigvee mn \leq \ell(\alpha)\bigvee g \in S_m(\mathrm{Sub}(\alpha))$$

y las hipótesis

$$h \in \mathrm{Fun}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner) \wedge \mathrm{rang}(h) = m \wedge \bigwedge i < m(g_i \in \mathrm{Fun}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner) \wedge \mathrm{rang}(g_i) = n),$$

pero comprimiendo toda esta información en una fórmula sólo volvemos ilegible una definición tan clara como 1.8.

Dejamos a cargo del lector formalizar (si lo cree imprescindible) la definición de los axiomas de los funtores definidos por recursión. El caso es que así podemos completar la definición de una fórmula $\alpha \in \mathrm{Ax}(\ulcorner\mathrm{ARP}\urcorner)$ que se corresponde con la definición 1.8 y en particular permite probar que si α es un axioma (metamatemático) de ARP, entonces

$$\underset{\mathrm{ARP}}{\vdash} \ulcorner\alpha\urcorner \in \mathrm{Ax}(\ulcorner\mathrm{ARP}\urcorner).$$

Deducciones Seguidamente, el lector debería convencerse de que no hay nada que impida formalizar en ARP la definición de deducción dada en 1.11. El resultado tiene que ser una fórmula

$$\Pi \underset{\ulcorner\mathrm{ARP}\urcorner}{\overset{d}{\vdash}} \alpha \equiv d_{\ell(d)\dot{-}1} = \alpha \wedge \bigwedge i < \ell(d)\,(d_i \in \mathrm{Form}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner) \wedge$$

$$(d_i \in \Pi \vee d_i \in \mathrm{Ax}(\ulcorner\mathrm{ARP}\urcorner) \vee \cdots)),$$

donde los puntos suspensivos representan cuatro fórmulas más en la disyunción que expresan que d_i se sigue de fórmulas precedentes por una de las cuatro reglas de inferencia de ARP, pero no ganamos nada condensando dichas reglas en fórmulas de ARP. Basta observar que para formalizarlas no tenemos que hacer referencia a nada que no podamos acotar en términos de la sucesión d o de sus términos.

Omitiremos el conjunto de premisas Π cuando sea vacío, es decir, que

$$\underset{\ulcorner\mathrm{ARP}\urcorner}{\overset{d}{\vdash}} \alpha \;\equiv\; \varnothing \underset{\ulcorner\mathrm{ARP}\urcorner}{\overset{d}{\vdash}} \alpha$$

y entonces diremos que d es una *demostración* de α en $\ulcorner\mathrm{ARP}\urcorner$.

En este punto nos encontramos con un hecho no trivial en cuanto a la formalización en ARP, y es que, en principio, no podemos definir una fórmula $\Pi \underset{\ulcorner\mathrm{ARP}\urcorner}{\vdash} \alpha$, omitiendo la deducción d mediante un cuantificador existencial, ya que no tenemos forma de acotarlo. Ahora bien, esto no impide formalizar en ARP todos los resultados expuestos en el capítulo I que no involucren el concepto de "verdad". Simplemente, en lugar de afirmar que "tal fórmula" es un teorema, tenemos que afirmar que "tal demostración demuestra tal fórmula". Por ejemplo, la formalización del teorema de deducción consiste en definir funtores D, P, D' de modo que

$$\Pi \cup \{\alpha\} \underset{\ulcorner\mathrm{ARP}\urcorner}{\overset{d}{\vdash}} \beta \to \left(\Pi \cup P(d,\alpha) \underset{\ulcorner\mathrm{ARP}\urcorner}{\overset{D(d,\alpha)}{\vdash}} \alpha \ulcorner\to\urcorner \beta\right) \wedge \bigwedge \gamma \in P(d,\alpha) \underset{\ulcorner\mathrm{ARP}\urcorner}{\overset{D'(d,\alpha,\gamma)}{\vdash}} \gamma$$

(con todas las hipótesis y conclusiones adicionales del teorema, es decir, hay que suponer que la deducción d no usa las reglas S_1 o I_0 respecto a variables que estén en α y podemos concluir que la deducción $D(d,\alpha)$ sólo usa estas reglas respecto de variables para las que ya se usaban en d).

Podemos asegurar que existen tales funtores (sin necesidad de definirlos explícitamente) porque la demostración del teorema de deducción es totalmente constructiva: determina completamente qué teoremas $P(d, \alpha)$ necesitamos tomar como premisas (y cuáles son sus demostraciones) y cómo construir una deducción de $\alpha \to \beta$ a partir de una deducción de β.

Con esta precaución de explicitar siempre las deducciones, es pura rutina convencerse de que todos los resultados sobre ARP demostrados en los capítulos precedentes son teoremas de ARP salvo los que involucran los conceptos de "denotación", "satisfacción" o "verdad", como es el caso del teorema 1.16.

3.2 Lenguajes formales de primer orden

La Aritmética Recursiva Primitiva es una teoría axiomática en la que podemos razonar formalmente, es decir, sin más precauciones que asegurarnos de que cuanto decimos puede expresarse mediante una fórmula de $\mathcal{L}_{\text{arp}}$ y que ésta puede demostrarse a partir de los axiomas de ARP sin más que aplicar mecánicamente las reglas de inferencia de ARP, sin plantearnos en ningún momento si las fórmulas consideradas son verdaderas o falsas (aunque con la garantía de que serán verdaderas si realmente logramos demostrarlas en ARP). Sin embargo, el razonamiento formal en ARP dista mucho de capturar plenamente lo que un matemático entiende por "razonar", por una parte porque ARP nos permite decir "para todo $x < y$" o "para todo $x \in y$", pero no podemos decir simplemente "para todo x", y lo mismo sucede con "existe un x". Un matemático no está acostumbrado a razonar con la obligación de acotar cada cuantificador que emplea. El segundo inconveniente de ARP es que no permite hablar más que de conjuntos finitos. A lo sumo, podemos hablar de colecciones infinitas a través de fórmulas, como $x \in \mathbb{Z}$ o $x \in \mathbb{Q}$, pero no nos permite hablar de conjuntos infinitos arbitrarios, como $\mathcal{P}\mathbb{Z}$ o $\mathbb{Z}^{\mathbb{Z}}$, por ejemplo. Los matemáticos tampoco están acostumbrados a que les digan: "de esto no puedes hablar".

Naturalmente, las limitaciones expresivas de ARP no son un defecto, sino una virtud, pues están pensadas para delimitar con precisión lo que podemos entender por "razonamiento estrictamente finitista", pero ahora nos vamos a ocupar de formalizar el razonamiento matemático propiamente dicho, sin limitaciones finitistas.

En cuanto al problema de cómo tratar formalmente con conjuntos infinitos, no nos vamos a ocupar ahora de él. Lo que haremos será exponer, no una teoría axiomática, sino una metateoría general sobre teorías axiomáticas con un aparato lógico común (el que, según hemos anunciado, capturará perfectamente lo que un matemático entiende por "razonamiento matemático"), de modo que podamos emplearlo para hablar de conjuntos infinitos, o de figuras geométricas, o de lo que se quiera, sin más que elegir oportunamente el lenguaje formal y los axiomas de partida.

En esta sección nos ocupamos del problema de definir una familia de lenguajes formales (compárese con los presentados en el capítulo I de [LM]) capaz de ajustarse a las necesidades expresivas de cualquier teoría matemática.

En particular, estos lenguajes formales dispondrán entre sus signos de los conectores lógicos $\neg, \to, \vee, \wedge, \leftrightarrow$ así como de cuantificadores $\bigwedge, \bigvee$ que podremos usar sin necesidad de acotarlos. Por razones técnicas, convendrá distinguir entre lenguajes con y sin igualador, porque en algunos resultados básicos convendrá considerar lenguajes sin igualador, pero a medio plazo trabajaremos únicamente con lenguajes con igualador.

También de forma opcional, incluiremos en los lenguajes formales un signo al que llamaremos descriptor, con el que hasta ahora no hemos tenido necesidad de tratar porque en ARP teníamos funtores para referirnos a cualquier objeto que podamos construir. Por ejemplo, si queremos referirnos al único número natural z que cumple $x + z = y$, usamos el funtor $y \dot{-} x$, pero vamos a tener ocasión de tratar con lenguajes dotados de pocos funtores y, entonces, si probamos que, para todo par de números naturales $x \leq y$, existe un único número natural z que cumple $x + z = y$ y queremos decir: "a ese número lo llamaremos $y \dot{-} x$", ¿cómo hay que entender esto si nuestro lenguaje no tiene un funtor $\dot{-}$? Lo que haremos será definir

$$y \dot{-} x \equiv z|(x + z = y).$$

El miembro derecho se lee: "el z tal que $x + z = y$". El signo $|$ es el descriptor y los términos definidos con él se llaman descripciones. En general, una expresión de la forma $x|\alpha$, donde α es una fórmula, se interpretará como "el único x que cumple α" cuando exista tal x, y el cálculo deductivo que vamos a exponer determinará cómo hay que entender las descripciones en los casos en los que no existe un único objeto que satisfaga la descripción (ya sea porque no hay ninguno o porque hay más de uno).

La definición siguiente puede entenderse como una definición informal en el mismo sentido que la definición 1.1, de modo que, como en el caso del lenguaje $\mathcal{L}_{\text{arp}}$, los signos de un lenguaje formal pueden ser cualquier cosa, desde signos gráficos hasta conceptos abstractos análogos a las piezas del ajedrez, pasando por números naturales, pero también podemos concebirla como una definición formalizada en ARP sin más que convenir en que los signos de un lenguaje formal son números naturales. Como esta definición contiene muchas ideas nuevas, vamos a omitir en ella los detalles técnicos sobre su formalización en ARP para concentrarnos en lo esencial y un poco más abajo incidiremos en dichos detalles.

Definición 3.2 Un *lenguaje formal de primer orden* $\mathcal{L}$ (con o sin igualador y, en caso de que tenga igualador, con o sin descriptor) es una colección de signos divididos en las categorías siguientes y de modo que cumplan las propiedades que se indican:

Variables libres Un lenguaje $\mathcal{L}$ debe tener infinitas variables libres. Cada variable libre debe tener asociado un número natural distinto al que llamaremos su *índice,* de tal forma que todo natural es índice de una variable libre de $\mathcal{L}$. Llamaremos x_i a la variable libre de índice i de $\mathcal{L}$.

Variables ligadas Un lenguaje $\mathcal{L}$ debe tener infinitas variables ligadas. Cada variable ligada debe tener asociado un número natural distinto al que llamaremos su *índice,* de tal forma que todo natural es índice de una variable ligada de $\mathcal{L}$. Llamaremos u_i a la variable ligada de índice i de $\mathcal{L}$.

Constantes Un lenguaje $\mathcal{L}$ puede tener cualquier cantidad de constantes, desde ninguna hasta infinitas. Si hay constantes, cada una de ellas deberá tener asociado un *índice* distinto, de modo que, llamando c_i a la constante de índice i, las constantes de $\mathcal{L}$ serán de la forma $c_0, \ldots, c_n$, si son un número finito, o bien $c_0, c_1, \ldots$ para todos los números naturales i.

Relatores Cada relator debe tener asociado un número natural no nulo al que llamaremos su *rango.* Llamaremos relatores n-ádicos a los relatores de rango n. El número de relatores n-ádicos de $\mathcal{L}$ puede variar entre uno e infinitos (pero tiene que haber al menos uno). Cada relator deberá tener asociado un *índice* en las mismas condiciones que las constantes.

Si $\mathcal{L}$ es un *lenguaje con igualador,* entonces tiene un relator diádico $=$ al que daremos un trato especial y al que llamaremos *igualador.*

Funtores Cada funtor ha de llevar asociado un *rango* y un *índice* en las mismas condiciones que los relatores.

Conectores Un lenguaje $\mathcal{L}$ debe tener cinco conectores lógicos, a los que llamaremos *negador* $\neg$, *implicador* $\rightarrow$, *disyuntor* $\vee$, *conjuntor* $\wedge$ y coimplicador $\leftrightarrow$.

Cuantificador universal Llamaremos $\bigwedge$ al *cuantificador universal* (o *generalizador*) de $\mathcal{L}$.

Cuantificador existencial Llamaremos $\bigvee$ al *cuantificador existencial* (o *particularizador*) de $\mathcal{L}$.

Descriptor Llamaremos $|$ al descriptor de $\mathcal{L}$ en caso de que lo tenga. (De ello depende que $\mathcal{L}$ sea un lenguaje con o sin descriptor.)

Cada signo de $\mathcal{L}$ debe pertenecer a una de estas categorías y sólo a una. Las constantes, los funtores y los relatores se llaman *signos eventuales* de $\mathcal{L}$, mientras que los restantes son *signos obligatorios.* (Entenderemos que el igualador es obligatorio en los lenguajes con igualador y que el descriptor es obligatorio en los lenguajes con descriptor.)

En la definición precedente mantenemos la distinción entre uso y mención que hemos establecido al presentar el lenguaje del cálculo proposicional y el de ARP. Así, por ejemplo, "$\bigvee$" no es el particularizador, sino el nombre que le damos al particularizador de cualquier lenguaje formal, el cual puede ser cualquier cosa razonable.

Veamos un ejemplo concreto:

El lenguaje de ARP$^+$ Extendamos el lenguaje $\mathcal{L}_{\text{arp}}$ añadiéndole un nuevo signo u y, llamemos *variables ligadas* a las sucesiones de signos formadas por u seguido de un numeral:

$$u_0 \equiv u0, \quad u_1 \equiv uS0, \quad u_2 \equiv uSS0, \quad \ldots$$

Ahora definimos $\mathcal{L}^+_{\text{arp}}$ como el lenguaje formal (con igualador y sin descriptor) cuyos signos son:

Variables libres La variable libre de $\mathcal{L}^+_{\text{arp}}$ de índice i es la variable x_i de $\mathcal{L}_{\text{arp}}$ (de modo que una cadena de signos de $\mathcal{L}_{\text{arp}}$ es considerada ahora como un único signo de $\mathcal{L}^+_{\text{arp}}$).

Variables ligadas La variable ligada de $\mathcal{L}^+_{\text{arp}}$ de índice i es la variable u_i de la extensión de $\mathcal{L}_{\text{arp}}$ que acabamos de considerar.[1]

Constantes El lenguaje $\mathcal{L}^+_{\text{arp}}$ tiene una única constante, que es el signo 0 de $\mathcal{L}_{\text{arp}}$.

Relatores El igualador de $\mathcal{L}^+_{\text{arp}}$ será el signo $=$ de $\mathcal{L}_{\text{arp}}$ y será, de hecho, el único relator de $\mathcal{L}^+_{\text{arp}}$.

Funtores El lenguaje $\mathcal{L}^+_{\text{arp}}$ tiene como funtores los funtores de $\mathcal{L}^+_{\text{arp}}$, con los mismos rangos (con lo que, nuevamente, estamos tomando como signos individuales de $\mathcal{L}^+_{\text{arp}}$ lo que son cadenas de signos del lenguaje $\mathcal{L}_{\text{arp}}$). El índice de cada funtor es el que hemos definido en la sección anterior en el apartado "enumeración de funtores" (en la página 120), de modo que el funtor de índice k es f_k.

Conectores y cuantificadores Tomamos signos cualesquiera, distintos de los precedentes, como conectores y cuantificadores.

■

Técnicamente, la distribución de los signos de un lenguaje formal en las distintas categorías que indica la definición es arbitraria (es decir, elegimos uno cualquiera como negador, otro cualquiera como implicador, otro lo tomamos como funtor triádico, etc.), pero la razón de fondo de esta definición se entiende al considerar la semántica asociada a los lenguajes formales. El primer paso en esta línea es introducir el concepto de modelo de un lenguaje formal.

[1] En el capítulo I de [LM] consideramos lenguajes formales con un único conjunto de variables, sin distinguir entre variables libres y ligadas. Veremos que la diferencia no es esencial, en el sentido de que se puede hacer lo mismo con los lenguajes formales que estamos definiendo aquí con esta distinción y los lenguajes definidos en [LM], pero la distinción entre variables libres y ligadas es necesaria para la aplicación de las técnicas del cálculo secuencial que exponemos en [CS].

Nota La definición siguiente no es formalizable en ARP (pero en el capítulo VII presentaremos una teoría donde sí que puede ser formalizada (definición 7.29)), por lo que, de momento, sólo es aplicable cuando consideramos lenguajes formales informalmente. Para dejar claro qué partes de la teoría son formalizables en ARP y cuáles —de momento— deben ser entendidas informalmente, en el segundo caso usaremos el tipo de letra que hemos empezado a usar en este párrafo. Conviene que el lector observe que si se salta los párrafos con este tipo de letra puede seguir leyendo los restantes sin que falte nada (salvo motivaciones para las definiciones). ∎

En este punto nos apoyaremos en las observaciones que hemos hecho al final de la introducción sobre el uso informal de los conceptos de "conjunto", "relación", "función", etc.

Definición 3.3 Un *modelo* M de un lenguaje formal $\mathcal{L}$ viene determinado por los elementos siguientes:

- Un conjunto no vacío (que representaremos también por M) al que llamaremos *universo* del modelo. Sus elementos serán los objetos de los que queremos hablar con el lenguaje $\mathcal{L}$. Por ejemplo, el *modelo natural* de $\mathcal{L}^+_{\mathrm{arp}}$ tiene por universo el conjunto $\mathbb{N}$ de todos los números naturales.

- Un criterio que asigne a cada constante c de $\mathcal{L}$ un objeto de M, que representaremos por $\bar{c}$ o por $M(c)$, y que será el objeto nombrado por la constante c. Por ejemplo, el modelo natural de $\mathcal{L}^+_{\mathrm{arp}}$ asigna a la constante 0 el número natural $\mathbb{N}(0) = 0$.

  Vemos así que las constantes de un lenguaje formal son los signos que usaremos para nombrar a objetos concretos, y cada modelo fija el objeto al que hace referencia cada constante.

- Un criterio que a cada relator n-ádico R de $\mathcal{L}$ le asigne una relación n-ádica en M que representaremos por $\bar{R}$ o por $M(R)$. Si $\mathcal{L}$ es un lenguaje con igualador, entonces la relación asociada al igualador $=$ será necesariamente la relación de identidad en M. Puesto que $\mathcal{L}^+_{\mathrm{arp}}$ no tiene más relator que el igualador, su interpretación $\mathbb{N}(=)$ está ya prefijada por la definición de modelo.

  Es en este sentido en el que el igualador es un relator especial: está pensado para hacer referencia a la relación de identidad, mientras que a cualquier otro relator le podemos dar cualquier interpretación cuando definamos un modelo.

  Así pues, los relatores son los signos que empleamos para hacer referencia a relaciones, igual que los funtores son los signos que empleamos para hacer referencia a funciones:

- Un criterio que a cada funtor n-ádico f de $\mathcal{L}$ le asigne una función n-ádica en M que representaremos por $\bar{f}$ o por $M(f)$.

  En el caso de $\mathcal{L}^+_{\mathrm{arp}}$, sus funtores son los mismos que los de $\mathcal{L}_{\mathrm{arp}}$, y establecemos que la interpretación $\mathbb{N}(f)$ de cada funtor f de $\mathcal{L}^+_{\mathrm{arp}}$ es la función recursiva primitiva $F(f)$ que le hemos asignado en la definición 1.3.

- En el caso en que $\mathcal{L}$ tenga descriptor, un elemento d de M al que llamaremos *descripción impropia* de M (será el elemento al que harán referencia las descripciones para las que no exista un único objeto en M que satisfaga la descripción). Esto no se aplica al caso de $\mathcal{L}^+_{\mathrm{arp}}$.

La definición de modelo no asigna ninguna interpretación a los conectores y cuantificadores porque éstos tendrán siempre la misma interpretación en todos los modelos (el negador significará siempre "no", etc.), por lo que no es la definición de modelo el lugar adecuado para especificarla.

Cadenas de signos Como en $\mathcal{L}_{\mathrm{arp}}$, dado cualquier lenguaje formal $\mathcal{L}$ de primer orden, podemos considerar cadenas de signos de $\mathcal{L}$, es decir, sucesiones finitas de signos de $\mathcal{L}$, con posibles repeticiones, en un orden dado. Usaremos la notación $\zeta_1 \equiv \zeta_2$ y $\zeta_1 \not\equiv \zeta_2$ para expresar que dos cadenas constan (o no) de los mismos signos en el mismo orden y, dadas unas cadenas de signos $\zeta_1, \ldots, \zeta_n$, podemos considerar su *yuxtaposición* $\zeta_1 \cdots \zeta_n$. ∎

Formalización en ARP Toda la parte sintáctica del estudio de los lenguajes de primer orden es puramente finitista y puede estudiarse sin necesidad de enmarcarla en ninguna teoría formal, pero —como ya hemos señalado— es, de hecho, formalizable en ARP. Por ello podemos considerar que todo lo que estamos diciendo en este tema sin relación con la semántica lo estamos definiendo y razonando en ARP, pero que trabajamos tal y como lo hacen habitualmente los matemáticos cuando trabajan en una teoría formal (usualmente ZFC), que no es escribir fórmulas del lenguaje formal correspondiente, sino hablar en una lengua natural (en este caso el castellano) dando por hecho que el lector sabrá identificar cada cosa que se diga con las fórmulas oportunas del lenguaje formal considerado, y no dar demostraciones detalladas, sino destacar únicamente las ideas relevantes de cada argumento.

Por ello no emplearemos signos lógicos en los enunciados y demostraciones, y por ello mismo no necesitaremos usar ángulos de Quine para referirnos a los signos de un lenguaje formal definido dentro de ARP. El único signo formal que usaremos habitualmente será el igualador, y por ello escribiremos $\zeta_1 \equiv \zeta_2$ para lo que, trabajando en ARP, es en realidad $\zeta_1 = \zeta_2$.

No obstante, en este apartado vamos a detallar la formalización de la definición de lenguaje formal, para que el lector acabe de formarse una idea clara de lo que esto supone.

Para definir un lenguaje formal de primer orden $\mathcal{L}$ en ARP hay que definir varias fórmulas y funtores que cumplan ciertas condiciones.

Variables libres Necesitamos una fórmula $x \in \mathrm{VarLib}(\ulcorner\mathcal{L}\urcorner)$ junto con dos funtores x, ind con los que podamos demostrar:

1. $x_i \in \mathrm{VarLib}(\ulcorner\mathcal{L}\urcorner)$,
2. $\mathrm{ind}(x_i) = i$,
3. $x \in \mathrm{VarLib}(\ulcorner\mathcal{L}\urcorner) \land i = \mathrm{ind}(x) \to x = x_i$.

En el caso de $\ulcorner\mathcal{L}^+_{\text{arp}}\urcorner$, tomamos $x \in \text{VarLib}(\ulcorner\mathcal{L}^+_{\text{arp}}\urcorner) \equiv x \in \text{Var}(\ulcorner\mathcal{L}_{\text{arp}}\urcorner)$, con los funtores x e ind que ya tenemos definidos para $\ulcorner\mathcal{L}_{\text{arp}}\urcorner$.

Variables ligadas Necesitamos una fórmula $u \in \text{VarLig}(\ulcorner\mathcal{L}\urcorner)$ junto con dos funtores u, ind que cumplan propiedades análogas a las anteriores. De hecho, el funtor ind puede ser el mismo (un mismo funtor puede asignar un índice tanto a las variables libres como a las variables ligadas).

En el caso de $\ulcorner\mathcal{L}^+_{\text{arp}}\urcorner$ podemos modificar la definición de $\ulcorner\mathcal{L}_{\text{arp}}\urcorner$ añadiendo el signo $\ulcorner u\urcorner = 8$ y esto permite definir la fórmula $u \in \text{VarLig}(\ulcorner\mathcal{L}\urcorner)$ y los funtores u, ind exactamente igual que en el caso de las variables libres (con el mismo funtor ind).

Constantes Necesitamos una fórmula $c \in \text{Const}(\ulcorner\mathcal{L}\urcorner)$ y dos funtores c, ind (y no perdemos generalidad si suponemos que el segundo es el mismo de los apartados anteriores) de modo que se cumplan las propiedades análogas a las de las variables en el caso de que $\ulcorner\mathcal{L}\urcorner$ vaya a tener infinitas constantes. Si no va a tener ninguna, podemos tomar $c \in \text{Const}(\ulcorner\mathcal{L}\urcorner) \equiv c \neq c$, mientras que si va a tener un número finito, necesitamos además un numeral NC de modo que los teoremas necesarios son:

1. $i < \text{NC} \to c_i \in \text{Const}(\ulcorner\mathcal{L}\urcorner)$,
2. $i < \text{NC} \to \text{ind}(c_i) = i$,
3. $c \in \text{Const}(\ulcorner\mathcal{L}\urcorner) \wedge i = \text{ind}(c) \to i < \text{NC} \wedge c = c_i$.

En el caso de $\ulcorner\mathcal{L}^+_{\text{arp}}\urcorner$, tomamos $c \in \text{Const}(\ulcorner\mathcal{L}^+_{\text{arp}}\urcorner) \equiv c = \ulcorner 0\urcorner$, $\text{NC} \equiv 1$, cualquier funtor que cumpla $c_0 = \ulcorner 0\urcorner$ y definimos ind de modo que $\text{ind}(\ulcorner 0\urcorner) = 0$. Claramente así se cumplen las condiciones anteriores.

Relatores Necesitamos una fórmula $R \in \text{Rel}(\ulcorner\mathcal{L}\urcorner)$ y funtores R, rang, ind de modo que, si hay infinitos relatores, podamos demostrar:

1. $R_i \in \text{Rel}(\ulcorner\mathcal{L}\urcorner)$,
2. $\text{ind}(R_i) = i$,
3. $\text{rang}(R_i) > 0$,
4. $R \in \text{Rel}(\ulcorner\mathcal{L}\urcorner) \wedge i = \text{ind}(R) \to R = R_i$.

Si el número de relatores tiene que ser finito, entonces necesitamos un numeral $\text{NR} \not\equiv 0$ de modo que los teoremas anteriores se restrinjan con la hipótesis $i < \text{NR}$ análogamente a como hemos hecho con las constantes.

En el caso de $\ulcorner\mathcal{L}^+_{\text{arp}}\urcorner$, basta tomar $R \in \text{Rel}(\ulcorner\mathcal{L}^+_{\text{arp}}\urcorner) \equiv R = \ulcorner{=}\urcorner$, con cualquier funtor R que cumpla $R_0 = \ulcorner{=}\urcorner$, cualquier funtor rang que cumpla $\text{rang}(\ulcorner{=}\urcorner) = 2$ y modificar el funtor ind para que cumpla $\text{ind}(\ulcorner{=}\urcorner) = 0$. Además $\text{NR} = 1$.

Funtores Necesitamos una fórmula $f \in \mathrm{Fun}(\ulcorner\mathcal{L}\urcorner)$ y funtores $f, \mathrm{rang}, \mathrm{ind}$ exactamente en las mismas condiciones del apartado anterior. Los funtores $\mathrm{rang}, \mathrm{ind}$ pueden ser los mismos sin pérdida de generalidad.

En el caso de $\ulcorner\mathcal{L}^+_{\mathrm{arp}}\urcorner$ tomamos $f \in \mathrm{Fun}(\ulcorner\mathcal{L}^+_{\mathrm{arp}}\urcorner) \equiv f \in \mathrm{Fun}(\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner)$ con los funtores f, rang, ind que definimos en la sección anterior (aunque la definición del índice se puede usar para modificar la definición del funtor ind de los apartados anteriores para que sobre los funtores coincida con el que ya tenemos definido, y así tenemos un único funtor índice para todos los casos).

Conectores y cuantificadores Necesitamos definir numerales distintos

$$\ulcorner\neg\urcorner, \quad \ulcorner\to\urcorner, \quad \ulcorner\vee\urcorner, \quad \ulcorner\wedge\urcorner, \quad \ulcorner\leftrightarrow\urcorner, \quad \ulcorner\bigwedge\urcorner, \quad \ulcorner\bigvee\urcorner.$$

En el caso de $\ulcorner\mathcal{L}^+_{\mathrm{arp}}\urcorner$, podemos tomar

$$\ulcorner\neg\urcorner = \langle 9\rangle\,, \quad \ulcorner\to\urcorner = \langle 10\rangle\,, \quad \ulcorner\vee\urcorner = \langle 11\rangle\,, \quad \ulcorner\wedge\urcorner = \langle 12\rangle\,, \quad \ulcorner\leftrightarrow\urcorner = \langle 13\rangle\,,$$
$$\ulcorner\bigwedge\urcorner = \langle 14\rangle\,, \quad \ulcorner\bigvee\urcorner = \langle 15\rangle\,.$$

Descriptor Si el lenguaje tiene descriptor, necesitamos especificar un numeral más $\ulcorner|\urcorner$.

Se tiene que poder demostrar que un mismo signo no es de dos tipos distintos, por ejemplo:

$$\neg(x \in \mathrm{VarLib}(\ulcorner\mathcal{L}\urcorner) \wedge x \in \mathrm{VarLig}(\ulcorner\mathcal{L}\urcorner)),$$

e igual con todas las parejas posibles para excluir así toda posible coincidencia.

Si tenemos definido en ARP un lenguaje formal $\ulcorner\mathcal{L}\urcorner$ que cumpla todas estas condiciones, podemos definir la fórmula

$$s \in \mathrm{Sig}(\ulcorner\mathcal{L}\urcorner) \equiv s \in \mathrm{VarLib}(\ulcorner\mathcal{L}\urcorner) \vee s \in \mathrm{VarLig}(\ulcorner\mathcal{L}\urcorner) \vee s \in \mathrm{Const}(\ulcorner\mathcal{L}\urcorner)$$
$$\vee\, s \in \mathrm{Rel}(\ulcorner\mathcal{L}\urcorner) \vee s \in \mathrm{Fun}(\ulcorner\mathcal{L}\urcorner) \vee s \in \{\ulcorner\neg\urcorner, \ulcorner\to\urcorner, \ulcorner\vee\urcorner, \ulcorner\wedge\urcorner, \ulcorner\leftrightarrow\urcorner, \ulcorner\bigwedge\urcorner, \ulcorner\bigvee\urcorner\}$$

que expresa que s es un signo de $\ulcorner\mathcal{L}\urcorner$ (añadiendo al final $\ulcorner|\urcorner$ si el lenguaje tiene descriptor), así como la fórmula

$$\zeta \in \mathrm{Cad}(\ulcorner\mathcal{L}\urcorner) \equiv \bigwedge i < \ell(\zeta)\ \zeta_i \in \mathrm{Sig}(\ulcorner\mathcal{L}\urcorner)$$

que expresa que ζ es una cadena de signos. A su vez, podemos considerar sucesiones ζ de cadenas de signos y las yuxtaposiciones $\zeta_1 \cdots \zeta_n$, exactamente igual que en el caso de ARP.

Para cada cadena de signos ζ, tenemos definido el conjunto $\mathrm{Sub}(\zeta)$ formado por las subcadenas de ζ (estrictamente contenidas en ζ) que es la base para definir funtores según el teorema 2.23, es decir, definiendo $F(\zeta)$ supuesto definido sobre las cadenas de $\mathrm{Sub}(\zeta)$. ∎

Expresiones Pasamos a definir las cadenas de signos de un lenguaje formal a las que atribuiremos un significado, pero cuidando mucho de hacerlo de forma puramente sintáctica, es decir, igual que lo hemos hecho en el caso de ARP, sin hacer referencia en ningún momento al significado pretendido. El lector encontrará en ella una primera explicación de la distinción que hemos establecido entre las variables libres y las ligadas:

Definición 3.4 Una cadena de signos θ de un lenguaje formal $\mathcal{L}$ es una *semiexpresión* si cumple una de las condiciones siguientes:[2]

1. θ es una variable (libre o ligada),
2. θ es una constante,
3. $\theta \equiv f^n t_1 \cdots t_n \equiv f^n(t_1, \ldots, t_n)$, donde f^n es un funtor n-ádico y $t_1, \ldots, t_n$ son semiexpresiones que empiecen por una variable, una constante, un funtor o (en su caso) el descriptor.
4. $\theta \equiv R^n t_1 \cdots t_n \equiv R^n(t_1, \ldots, t_n)$, donde R^n es un relator n-ádico y $t_1, \ldots, t_n$ son semiexpresiones en las mismas condiciones del apartado anterior.
5. θ es de una de las formas siguientes:

$$\neg\alpha, \quad \vee\alpha\beta \equiv (\alpha \vee \beta), \quad \rightarrow\alpha\beta \equiv (\alpha \rightarrow \beta),$$

$$\wedge\alpha\beta \equiv (\alpha \wedge \beta), \quad \leftrightarrow\alpha\beta \equiv (\alpha \leftrightarrow \beta),$$

   donde α, β, son dos semiexpresiones que empiecen por un relator, un conector o un cuantificador.

6. $\theta \equiv \bigwedge u\alpha$ o $\theta \equiv \bigvee u\alpha$ o (si $\mathcal{L}$ tiene descriptor) $\theta \equiv u|\alpha$, donde u es una variable ligada y α es una semiexpresión en las mismas condiciones del apartado anterior.

Las semiexpresiones que empiezan por una variable, un funtor o (en su caso) el descriptor se llaman *semitérminos,* mientras que las que empiezan por un relator, un conector o un cuantificador se llaman *semifórmulas.*

Para aligerar los enunciados, convendremos en que x, y, z representarán siempre variables libres, u, v, w variables ligadas, c una constante, f^n un funtor n-ádico, R^n un relator n-ádico, t un semitérmino, α una semifórmula y θ una semiexpresión. En estos términos, la definición precedente queda así:

Teorema 3.5 *Si θ es una semiexpresión de un lenguaje formal $\mathcal{L}$, se encuentra necesariamente en uno de los casos siguientes:*

[2]El lector debe reconocer aquí una definición por recurrencia, en la que se define un funtor F que vale 1 sobre las semiexpresiones y 0 en las sucesiones que no lo son, de modo que definimos si una cadena θ es una semiexpresión supuesto que ya hemos definido qué cadenas de $\mathrm{Sub}(t)$ son semiexpresiones.

1. *θ es una variable (libre o ligada) o una constante, en cuyo caso es un semitérmino.*

2. *$\theta \equiv f^n(t_1, \ldots, t_n)$, en cuyo caso también es un semitérmino.*

3. *$\theta \equiv R^n(t_1, \ldots, t_n)$, en cuyo caso también es una semifórmula.*

4. *$\theta \equiv \neg\alpha,\ \alpha \vee \beta,\ \alpha \to \beta,\ \alpha \wedge \beta,\ \alpha \leftrightarrow \beta$, en cuyo caso es una semifórmula.*

5. *$\theta \equiv \bigwedge u\, \alpha,\ \bigvee u\, \alpha$, en cuyo caso es una semifórmula.*

6. *$\theta \equiv u|\alpha$, en cuyo caso es un semitérmino.*

Además, en los casos 2 y 3, los semitérminos $t_1, \ldots, t_n$ están unívocamente determinados[3] *por θ, al igual que las semifórmulas α, β en 4.*

Observemos que en lenguajes sin descriptor podemos definir primero los semitérminos y luego las semifórmulas, pero si hay descriptor hay que definir ambos conceptos simultáneamente porque los relatores construyen semifórmulas a partir de semitérminos y el descriptor construye semitérminos a partir de semifórmulas.

Cuando haya que escribir varios cuantificadores seguidos del mismo tipo, como $\bigvee u \bigvee v \bigvee w$ escribiremos más brevemente $\bigvee uvw$.

En los lenguajes con igualador adoptaremos también la notación

$$(t_1 = t_2) \equiv {=}t_1t_2, \qquad t_1 \neq t_2 \equiv \neg(t_1 = t_2).$$

Al tratar con lenguajes formales particulares adoptaremos tácitamente convenios de notación similares. Por ejemplo, en el lenguaje de ARP^+ escribiremos $(t_1 + t_2) \equiv {+}t_1t_2$. ∎

Desde un punto de vista semántico, los semitérminos son cadenas de signos que nombran objetos y las semifórmulas las cadenas de signos que afirman algo. Esto se plasma en la definición de denotación y satisfacción en un modelo:

Definición 3.6 Si M es un modelo de un lenguaje formal $\mathcal{L}$, una *valoración* en M es cualquier criterio v que a cada variable x de $\mathcal{L}$ (libre o ligada) le asigna un elemento $v(x)$ del universo M del modelo.

Si a es un objeto del universo del modelo, llamaremos v^a_x a la valoración que actúa como v salvo que a x le asigna el objeto a.

Definimos el *objeto denotado* por un semitérmino t en un modelo M respecto de una valoración v (al que representaremos por $M(t)[v]$) y la *satisfacción* de una semifórmula en un modelo M respecto de una valoración v (que representaremos por $M \vDash \alpha[v]$) mediante las condiciones siguientes:

[3] Esto se prueba definiendo un funtor que a cada sucesión le asigna su semiexpresión inicial, si es que la tiene, y 0 en caso contrario.

1. Si x es una variable (libre o ligada), $M(x)[v] = v(x)$.

2. $M(c)[v] = M(c)$ es el objeto asociado a la constante c por el modelo M.

3. $M(f^n(t_1, \ldots, t_n)) = M(f^n)(M(t_1)[v], \ldots, M(t_n)[v])$, es decir, el objeto denotado por el semitérmino $f^n(t_1, \ldots, t_n)$ se calcula como la imagen por la función $M(f^n)$ asociada al funtor f^n por el modelo de los objetos denotados por los semitérminos $t_1, \ldots, t_n$.

4. $M \vDash R^n(t_1, \ldots, t_n)$ si y sólo si $M(R^n)(M(t_1)[v], \ldots, M(t_n)[v])$, de modo que la semifórmula $R^n(t_1, \ldots, t_n)$ es satisfecha si y sólo si los objetos denotados por los semitérminos $t_1, \ldots, t_n$ cumplen la relación $M(R^n)$ asociada al relator por el modelo.

5. $M \vDash \neg\alpha[v]$ si y sólo si no $M \vDash \alpha[v]$.

6. $M \vDash (\alpha \vee \beta)[v]$ si y sólo si $M \vDash \alpha[v]$ o $M \vDash \beta[v]$.

7. $M \vDash (\alpha \to \beta)[v]$ si y sólo si no $M \vDash \alpha[v]$ o $M \vDash \beta[v]$.

8. $M \vDash (\alpha \wedge \beta)[v]$ si y sólo si $M \vDash \alpha[v]$ y $M \vDash \beta[v]$.

9. $M \vDash (\alpha \leftrightarrow \beta)[v]$ si y sólo si $M \vDash \alpha[v]$ y $M \vDash \beta[v]$ o bien no $M \vDash \alpha[v]$ y no $M \vDash \beta[v]$.

10. $M \vDash \bigwedge u\alpha[v]$ si y sólo si, para todo objeto a del universo de M, se cumple $M \vDash \alpha[v^a_u]$.

11. $M \vDash \bigvee u\alpha[v]$ si y sólo si, existe un objeto a del universo de M tal que $M \vDash \alpha[v^a_u]$.

12. $M(u|\alpha)[v]$ es el único objeto a de M que cumple $M \vDash \alpha[v^a_u]$ si existe tal objeto, o bien la descripción impropia del modelo si no existe tal objeto (sea porque no existe ninguno, sea porque hay más de uno).

Es en esta definición[4] donde establecemos el significado pretendido de los conectores, de los cuantificadores y del descriptor.

[4]Es frecuente que cuando un formalista lee en un libro definiciones como la de "modelo", que hemos dado previamente, o las de denotación y satisfacción, que acabamos de dar, acabe persuadido de que se trata de matemática formal, pero que el autor esconde hipócritamente los axiomas que está suponiendo. Ciertamente, estas definiciones son formalizables en cualquier teoría de conjuntos, e incluso —según veremos— en teorías mucho más débiles. Sin embargo, esto no contradice que, en este momento en el que no disponemos de una teoría axiomática capaz de formalizar estos conceptos, podamos emplearlos informalmente, sin suponer ningún axioma. Simplemente estableciendo un universo M intuitivamente bien definido y asignando a cada fórmula α de un lenguaje formal dado una afirmación $M \vDash \alpha[v]$ sobre los objetos de M. Es un hecho que podemos hacer esto, independientemente de si puede probarse a partir de unos u otros axiomas.

Variables libres y ligadas En una semifórmula como $\bigwedge u\, Ruv$, la variable ligada u aparece vinculada al generalizador, mientras que la variable ligada v no está vinculada a ningún cuantificador. Vamos a definir las expresiones como las semiexpresiones para las que esto no sucede, para lo cual tenemos que precisar esta idea de "estar vinculado a un cuantificador":

Definición 3.7 Sea $\mathcal{L}$ un lenguaje formal. Definimos como sigue el conjunto de variables que *aparecen libres* en una semiexpresión de $\mathcal{L}$:

1. La única variable que aparece libre en una variable x (libre o ligada) es x.
2. En una constante c no aparecen variables libres.
3. Las variables que aparecen libres en un semitérmino $f^n(t_1, \ldots, t_n)$ son las que aparecen libres en alguno de los términos t_i.
4. Las variables que aparecen libres en una semifórmula $R^n(t_1, \ldots, t_n)$ son las que aparecen libres en alguno de los términos t_i.
5. Las variables que aparecen libres en $\neg\alpha$ son las mismas que aparecen libres en α.
6. Las variables que aparecen libres en $\alpha \to \beta$, $\alpha \vee \beta$, $\alpha \wedge \beta$ o $\alpha \leftrightarrow \beta$ son las que aparecen libres en α y las que aparecen libres en β.
7. Las variables que aparecen libres en $\bigwedge u\, \alpha$ o $\bigvee u\, \alpha$ o (en su caso) en $u|\alpha$ son las que aparecen libres en α y son distintas de u.

Similarmente definimos el conjunto de las variables que aparecen ligadas en una semiexpresión de $\mathcal{L}$:

Definición 3.8 Sea $\mathcal{L}$ un lenguaje formal. Definimos como sigue el conjunto de variables que *aparecen ligadas* en una semiexpresión de $\mathcal{L}$:

1. En una variable x (libre o ligada) no aparecen variables ligadas.
2. En una constante c no aparecen variables ligadas.
3. Las variables que aparecen ligadas en un semitérmino $f^n(t_1, \ldots, t_n)$ son las que aparecen ligadas en alguno de los términos t_i.
4. Las variables que aparecen ligadas en una semifórmula $R^n(t_1, \ldots, t_n)$ son las que aparecen ligadas en alguno de los términos t_i.
5. Las variables que aparecen ligadas en $\neg\alpha$ son las mismas que aparecen ligadas en α.
6. Las variables que aparecen ligadas en $\alpha \to \beta$, $\alpha \vee \beta$, $\alpha \wedge \beta$ o $\alpha \leftrightarrow \beta$ son las que aparecen ligadas en α y las que aparecen ligadas en β.
7. Las variables que aparecen ligadas en $\bigwedge u\, \alpha$ o $\bigvee u\, \alpha$ o (en su caso) en $u|\alpha$ son u más las que aparecen ligadas en α.

De acuerdo con estas definiciones, las variables que aparecen ligadas en una semiexpresión son siempre variables ligadas, pero las variables que aparecen libres pueden ser libres o ligadas.

Llamaremos *expresiones, términos y fórmulas,* respectivamente a las semiexpresiones, semitérminos y semifórmulas en las que ninguna variable ligada aparece libre o, equivalentemente, son las semiexpresiones en las que las variables que aparecen libres son libres y las que aparecen ligadas son ligadas.

Una expresión es *abierta* si tiene variables libres. En caso contrario es cerrada. Un *designador* es un término cerrado. Una *sentencia* es una fórmula cerrada.

Nuestra intención es trabajar únicamente con términos y fórmulas, pero sucede que para construir las expresiones necesitamos usar semiexpresiones. Por ejemplo, esto es una fórmula de ARP^+:

$$\bigwedge uv(u + v = v + u),$$

que está construida a partir de los semitérminos u, v, $u + v$, $v + u$ y de las semifórmulas $u + v = v + u$ y $\bigwedge v(u + v = v + u)$. Sólo cuando añadimos el último cuantificador pasamos a tener una fórmula. ∎

Existencia con unicidad Para lenguajes $\mathcal{L}$ con igualador, definimos

$$\overset{1}{\bigvee} u\alpha \equiv \bigvee v \bigwedge u(\alpha \leftrightarrow u = v),$$

donde v es cualquier variable ligada distinta de u y que no esté en α.

Si M es un modelo de $\mathcal{L}$ y w es una valoración, se cumple que $M \vDash \overset{1}{\bigvee}\alpha[w]$ si y sólo si existe un único a en M tal que $M \vDash \alpha[w^a_u]$.

En efecto, $M \vDash \overset{1}{\bigvee}\alpha[w]$ si y sólo si existe un a en M tal que

$$M \vDash \bigwedge u(\alpha \leftrightarrow u = v)[w^a_v],$$

y esto sucede si y sólo si para, todo b en M, se cumple

$$M \vDash (\alpha \leftrightarrow u = v)[w^{ab}_{vu}],$$

que a su vez equivale a que, para todo b en M, se cumple $M \vDash \alpha[w^b_u]$ si y sólo si $b = a$, y esto significa que a es el único elemento de M que cumple $M \vDash \alpha[w^a_u]$. ∎

Fórmulas verdaderas y falsas Para definir el concepto de fórmula verdadera o falsa necesitamos el teorema siguiente, que afirma que la denotación o satisfacción de una expresión respecto de una valoración depende únicamente de cómo actúa la valoración sobre las variables que aparecen libres en ella:

Teorema 3.9 *Si M es un modelo de un lenguaje formal $\mathcal{L}$ y v, w son dos valoraciones que coinciden sobre las variables que aparecen libres en una semiexpresión θ, entonces, si θ es un semitérmino, se cumple que $M(\theta)[v] = M(\theta)[w]$, y si θ es una semifórmula, entonces $M \vDash \theta[v]$ si y sólo si $M \vDash \theta[w]$.*

DEMOSTRACIÓN: Si θ es una semiexpresión que incumple el teorema, podemos pasar a una subsemiexpresión[5] de longitud mínima que lo incumpla. Vamos a distinguir todos los casos posibles y veremos que en cualquiera de ellos llegamos a una contradicción.

No puede ocurrir que θ sea una variable x, pues entonces estaría libre en x y tendríamos que

$$M(\theta)[v] = v(\theta) = w(\theta) = M(\theta)[w].$$

Tampoco puede ocurrir que θ sea una constante c, pues entonces

$$M(\theta)[v] = M(c) = M(\theta)[w].$$

Si $\theta \equiv f^n(t_1, \ldots, t_n)$, entonces los términos t_i cumplirían el teorema (por la minimalidad de θ) y v y w coinciden en las variables que aparecen libres en ellos, luego

$$M(\theta)[v] = M(f)(M(t_1)[v], \ldots, M(t_n)[v])$$
$$= M(f)(M(t_1)[w], \ldots, M(t_n)[w]) = M(\theta)[w].$$

Si $\theta \equiv R^n(t_1, \ldots, t_n)$, tenemos que v y w coinciden en las variables libres en cada t_i, luego

$$M \vDash \theta[v] \text{ si y sólo si } M(R^n)(M(t_1)[v], \ldots, M(t_n)[v])$$

$$\text{si y sólo si } M(R^n)(M(t_1)[w], \ldots, M(t_n)[w]) \text{ si y sólo si } M \vDash \theta[w].$$

Si $\theta \equiv \neg\alpha$, $\alpha \vee \beta$, $\alpha \to \beta$, $\alpha \wedge \beta$, $\alpha \leftrightarrow \beta$, como v y w coinciden en las variables que aparecen libres en α y β (pues todas ellas aparecen libres en θ), el hecho de que α y β cumplan el teorema implica claramente que θ también lo cumple.

Supongamos ahora que $\theta \equiv \bigwedge u\alpha$. Entonces las variables libres de α son las de θ y tal vez también u, luego v y w coinciden en todas ellas salvo a lo sumo en u. Pero si a es un elemento de M, entonces v_u^a y w_u^a coinciden en todas las variables libres de α. Como α tiene longitud menor, tenemos que

$$M \vDash \theta[v] \text{ si y sólo si para todo } a \text{ en } M \ M \vDash \alpha[v_u^a] \text{ si y sólo si}$$

$$\text{para todo } a \text{ en } M \ M \vDash \alpha[w_u^a] \text{ si y sólo si } M \vDash \theta[w].$$

Si $\theta \equiv \bigvee u\alpha$, el razonamiento es análogo.

Por último, si $\theta \equiv u|\alpha$, como en el caso anterior, para todo a en M, las valoraciones v_u^a y w_u^a coinciden en todas las variables que aparecen libres en α, luego $M \vDash \alpha[v_u^a]$ si y sólo si $M \vDash \alpha[w_u^a]$. Por lo tanto, existe un único a en M que cumple una de las condiciones si y sólo si existe un único a en M que cumple la otra (y es el mismo objeto en ambos casos). Esto implica que $M(u|\alpha)[v] = M(u|\alpha)[w]$, sea porque ambos son el único a que cumple la condición, sea porque ambos son la descripción impropia fijada por M. ∎

Este teorema da pleno sentido a la definición siguiente:

[5] Un elemento de Sub(θ).

Definición 3.10 Si M es un modelo de un lenguaje formal $\mathcal{L}$, diremos que una semifórmula α es *verdadera* en M si $M \vDash \alpha[v]$ para toda valoración v en M (y lo representaremos por $M \vDash \alpha$), y diremos que α es *falsa* en M si no existe ninguna valoración en M respecto a la cual $M \vDash \alpha[v]$.

Como en el caso de ARP, aquí es clave que, aunque no sabríamos dar un sentido a que *todas* las valoraciones cumplan $M \vDash \alpha[v]$, por el teorema anterior, esto tiene que cumplirse para todas las asignaciones posibles de valores en M sobre el conjunto finito de las variables libres en α, de modo que podemos enumerar todas las posibilidades, lo que da sentido a la definición.

En ella estamos introduciendo también el mismo convenio que hemos empleado en ARP, según el cual una semifórmula α es verdadera si se cumple para cualquier forma de interpretar sus variables libres.

Si Γ es un conjunto de fórmulas de $\mathcal{L}$, escribiremos $M \vDash \Gamma$ (o, explícitamente, $M \vDash \gamma_1, \ldots, \gamma_n$) para indicar que todas las fórmulas de Γ son verdaderas en M.

Sustitución de variables por términos Generalizamos ahora el concepto de sustitución que hemos definido en ARP:

Definición 3.11 Diremos que una variable x (libre o ligada) *puede sustituirse* por un semitérmino t en una semiexpresión θ si se cumplen las condiciones siguientes:

1. Si θ es una variable o una constante, la sustitución es posible.
2. Si $\theta \equiv f^n(t_1, \ldots, t_n)$, la sustitución es posible si x puede sustituirse por t en cada uno de los semitérminos t_i.
3. Si $\theta \equiv R^n(t_1, \ldots, t_n)$, la sustitución es posible si x puede sustituirse por t en cada uno de los semitérminos t_i.
4. Si $\theta \equiv \neg\alpha$, $\alpha \vee \beta$, $\alpha \to \beta$, $\alpha \wedge \beta$, $\alpha \leftrightarrow \beta$, la sustitución es posible si x puede sustituirse por t en α y (en su caso) en β.
5. Si $\theta \equiv \bigwedge u\alpha$, $\bigvee u\alpha$, $u|\alpha$, la sustitución es posible salvo si $x \not\equiv u$ y no se puede sustituir x por t en α o bien la variable u aparece libre en t.

En particular, es claro que siempre podemos sustituir cualquier variable por cualquier término en cualquier semiexpresión. El único impedimento posible es que en t aparezca libre una variable ligada u que aparezca ligada en θ.

Si x es una variable que puede sustituirse por un semitérmino t en una semiexpresión θ, definimos la *sustitución* de x por t en θ como la semiexpresión $\mathtt{S}^t_x\theta$ definida por las reglas siguientes:

1. $\mathtt{S}^t_x x_0 \equiv \begin{cases} t & \text{si } x \equiv x_0, \\ x_0 & \text{si } x \not\equiv x_0. \end{cases}$
2. $\mathtt{S}^t_x c \equiv c$.

3. $\mathtt{S}^t_x f^n(t_1,\ldots,t_n) \equiv f^n(S^t_x t_1,\ldots,S^t_x t_n)$.
4. $\mathtt{S}^t_x R^n(t_1,\ldots,t_n) \equiv R^n(S^t_x t_1,\ldots,S^t_x t_n)$.
5. $\mathtt{S}^t_x \neg\alpha \equiv \neg \mathtt{S}^t_x \alpha$.
6. $\mathtt{S}^t_x(\alpha \vee \beta) \equiv \mathtt{S}^t_x\alpha \vee \mathtt{S}^t_x\beta$.
7. $\mathtt{S}^t_x(\alpha \to \beta) \equiv \mathtt{S}^t_x\alpha \to \mathtt{S}^t_x\beta$.
8. $\mathtt{S}^t_x(\alpha \wedge \beta) \equiv \mathtt{S}^t_x\alpha \wedge \mathtt{S}^t_x\beta$.
9. $\mathtt{S}^t_x(\alpha \leftrightarrow \beta) \equiv \mathtt{S}^t_x\alpha \leftrightarrow \mathtt{S}^t_x\beta$.
10. $\mathtt{S}^t_x \bigwedge u\,\alpha \equiv \begin{cases} \bigwedge u\,\alpha & \text{si } u \equiv x,\\ \bigwedge u\,\mathtt{S}^t_x\,\alpha & \text{si } u \not\equiv x.\end{cases}$
11. $\mathtt{S}^t_x \bigvee u\,\alpha \equiv \begin{cases} \bigvee u\,\alpha & \text{si } u \equiv x,\\ \bigvee u\,\mathtt{S}^t_x\,\alpha & \text{si } u \not\equiv x.\end{cases}$
12. $\mathtt{S}^t_x u|\alpha \equiv \begin{cases} u|\alpha & \text{si } u \equiv x,\\ u|\mathtt{S}^t_x\,\alpha & \text{si } u \not\equiv x.\end{cases}$

En definitiva, $\mathtt{S}^t_x\theta$ no es sino la semiexpresión que resulta quitar cada aparición libre en θ de la variable x y poner en su lugar todo el semitérmino t, pero, como muestran las condiciones 10, 11 y 12, las variables que aparecen ligadas no se sustituyen. En particular, si x no está libre en θ, se cumple que $\mathtt{S}^t_x\theta \equiv \theta$.

Nota La razón por la que no admitimos que en ciertos casos una variable se sustituya por un semitérmino en una semiexpresión la ilustran los ejemplos siguientes: En $\mathcal{L}^+_{\mathrm{arp}}$ tenemos que

$$\mathtt{S}^v_y \bigvee u(y = x + u) \equiv \bigvee u(v = x + u),$$

$$\mathtt{S}^u_y \bigvee u(y = x + u) \equiv \bigvee u(u = x + u),$$

aunque en realidad la segunda sustitución no es válida, porque la variable ligada u aparece libre en el semitérmino por el que sustituimos x. Esto se traduce en que en el primer caso obtenemos una semifórmula que afirma que $x \leq v$ (lo que cabía esperar), mientras que en el segundo la fórmula resultante no significa que $x \leq u$, sino que $x = 0$. Esto se debe a que la variable u, que estaba libre, ha quedado ligada tras la sustitución, pero la segunda sustitución la hemos excluido en la definición. ∎

El teorema siguiente muestra que, con las restricciones que hemos impuesto, la sustitución se comporta siempre de forma razonable:

Teorema 3.12 *Si v es una valoración en un modelo M de un lenguaje formal $\mathcal{L}$, θ es una semiexpresión de $\mathcal{L}$, x es una variable (libre o ligada) y t es un semitérmino tal que x puede sustituirse por t en θ, entonces:*

1. *Si θ es un término, $M(\mathtt{S}^t_x\theta)[v] = M(\theta)[v_x^{M(t)[v]}]$.*
2. *Si θ es una fórmula, $M \vDash S^t_x\theta[v]$ si y sólo si $M \vDash \theta[v_x^{M(t)[v]}]$.*

DEMOSTRACIÓN: Como de costumbre, vamos a ver que no puede haber una semiexpresión θ mínima que incumpla el teorema, es decir, suponemos que se cumple para semiexpresiones menores y veremos que también se cumple para θ.

Si θ es una variable, distinguimos dos casos, según si $\theta \equiv x$ o no. En el primer caso,

$$M(\mathtt{S}^t_x\theta)[v] = M(t)[v] = M(\theta)[v_x^{M(t)[v]}].$$

En el segundo caso

$$M(\mathtt{S}^t_x\theta)[v] = M(\theta)[v] = v(\theta) = v_x^{M(t)[v]}(\theta) = M(\theta)[v_x^{M(t)[v]}].$$

Si θ es una constante o empieza por un funtor la conclusión es inmediata.

Si $\theta \equiv R^n(t_1, \ldots, t_n)$, estamos suponiendo que x se puede sustituir por t en cada semitérmino, y por la minimalidad de θ todos los t_i cumplen el teorema. Entonces:

$$M \vDash (\mathtt{S}^t_x\theta)[v] \text{ si y sólo si } M(R^n)(M(\mathtt{S}^t_x t_1)[v], \ldots, M(\mathtt{S}^t_x t_n)[v])$$

$$\text{si y sólo si } M(R^n)(M(t_1)[v_x^{M(t)[v]}], \ldots, M(t_n)[v_x^{M(t)[v]}])$$

$$\text{si y sólo si } M \vDash \theta[v_x^{M(t)[v]}].$$

Si $\theta \equiv \alpha \vee \beta$, entonces

$$M \vDash (\mathtt{S}^t_x\theta)[v] \text{ si y sólo si } M \vDash (\mathtt{S}^t_x\alpha)[v] \text{ o } M \vDash (\mathtt{S}^t_x\beta)[v]$$

si y sólo si $M \vDash \alpha[v_x^{M(t)[v]}]$ o $M \vDash \beta[v_x^{M(t)[v]}]$ si y sólo si $M \vDash \theta[v_x^{M(t)[v]}]$.

Los demás casos en los que θ se construye con un conector se razonan análogamente. Supongamos ahora que $\theta \equiv \bigwedge u\alpha$, y distinguimos dos casos: si $x \equiv u$, tenemos que

$$M \vDash (\mathtt{S}^t_x \bigwedge u\, \alpha)[v] \text{ si y sólo si } M \vDash (\bigwedge u\alpha)[v] \text{ si y sólo si } M \vDash (\bigwedge u\alpha)[v_x^{M(t)[v]}],$$

ya que, como x no está libre en $\bigwedge u\alpha$, el cambio en la valoración no afecta a la satisfacción. Si $x \not\equiv u$ (y entonces u no está libre en t, para que la sustitución sea posible),

$$M \vDash (\mathtt{S}^t_x\theta)[v] \text{ si y sólo si } M \vDash (\bigwedge u\, \mathtt{S}^t_x\alpha)[v] \text{ si y sólo si, para todo } a \text{ en } M,$$

$$M \vDash (\mathtt{S}^t_x\alpha)[v^a_u] \text{ si y sólo si, para todo } a \text{ en } M,\ M \vDash \alpha[v_{ux}^{aM(t)[v^a_u]}].$$

Ahora observamos que, como u no aparece libre en t, $M(t)[v^a_u] = M(t)[v]$. Por otro lado, como $x \not\equiv u$, podemos intercambiar u y x en la modificación de v, con lo que

$$M \vDash (\mathtt{S}^t_x\theta)[v] \text{ si y sólo si, para todo } a \text{ en } M,\ M \vDash \alpha[v_{xu}^{M(t)[v]a}]$$

$$\text{si y sólo si } M \vDash \theta[v_x^{M(t)[v]}].$$

Los casos $\theta \equiv \bigvee u\alpha$ y $\theta \equiv u|\alpha$ son análogos. ∎

Usaremos los mismos convenios de notación que en ARP para indicar las sustituciones. Concretamente, escribiremos $\theta(x_1,\ldots,x_n)$ para indicar que

$$\theta(t_1,\ldots,t_n) \equiv \mathtt{S}^{t_1}_{z_1}\cdots\mathtt{S}^{t_n}_{z_n}\mathtt{S}^{z_1}_{x_1}\cdots\mathtt{S}^{z_n}_{x_n}\theta,$$

donde $z_1,\ldots,z_n$ son variables (libres) que no estén ni en $t_1,\ldots,t_n$, ni en θ, ni en $x_1,\ldots,x_n$.

De este modo $\theta(t_1,\ldots,t_n)$ no es más que la expresión que resulta de sustituir cada variable x_i libre en θ por el término t_i. En particular, la elección de las variables auxiliares z_i es irrelevante. Sólo las consideramos para evitar que si, por ejemplo, en t_n aparece la variable x_{n-1}, ésta sea sustituida por t_{n-1} dentro de t_n cuando se efectúa la sustitución de x_{n-1} en θ tras haber sustituido x_n por t_n, pero todas las variables auxiliares acaban desapareciendo.

No hay que entender, salvo que se diga explícitamente, que las variables libres $x_1,\ldots,x_n$ aparezcan en θ, ni que no puedan aparecer en θ otras variables libres distintas de las indicadas. Escribir $\theta(x_1,\ldots,x_n)$ simplemente nos indica qué variables hay que sustituir por $t_1,\ldots,t_n$ para calcular $\theta(t_1,\ldots,t_n)$.

3.3 La lógica de primer orden

Nos ocupamos ahora de mostrar cómo se puede razonar formalmente con las fórmulas de un lenguaje formal de primer orden arbitrario. Esta sección y las siguientes se corresponden con el capítulo II de [LM]. Empezamos introduciendo algunos conceptos generales que ya conocemos para el caso de la aritmética recursiva primitiva:

Definición 3.13 Un *sistema deductivo formal* (de primer orden) F sobre un lenguaje formal $\mathcal{L}$ viene determinado por[6] un conjunto de fórmulas de $\mathcal{L}$, llamadas *axiomas* de F, y un conjunto de *reglas primitivas de inferencia* de F, que determinan cuándo una fórmula de $\mathcal{L}$ es *consecuencia inmediata* de otra u otras fórmulas de $\mathcal{L}$.

Una *deducción* en un sistema deductivo formal F a partir de un conjunto de fórmulas Γ es una sucesión finita $\alpha_1,\ldots,\alpha_n$ de fórmulas de $\mathcal{L}$ tales que cada α_i es un axioma de F, una fórmula de Γ o una consecuencia inmediata de fórmulas anteriores de la sucesión. Las fórmulas de Γ se llaman *premisas* de la deducción. Una *demostración* es una deducción sin premisas.

Una fórmula α es una *consecuencia* en F de un conjunto de fórmulas Γ si α es la última fórmula de una deducción en F a partir de Γ. Lo representaremos con la notación[7] $\Gamma \underset{F}{\vdash} \alpha$ o, más explícitamente: $\gamma_1,\ldots,\gamma_n \underset{F}{\vdash} \alpha$.

[6] La definición de sistema deductivo formal no es formalizable en ARP, en el sentido de que no podemos definir una fórmula que signifique "F es un sistema deductivo formal", pero si F es un sistema deductivo formal razonable, podremos formalizarlo en ARP en el sentido de que podremos definir una fórmula $\alpha \in \mathrm{Ax}(\ulcorner F\urcorner)$ que exprese que α es un axioma de $\ulcorner F\urcorner$ y una fórmula $\Gamma \underset{F}{\vdash} \alpha$ que exprese que la fórmula α es consecuencia inmediata del conjunto de fórmulas Γ.

[7] Esto es claramente formalizable en ARP salvo por el hecho de que no podemos definir una fórmula $\Gamma \underset{F}{\vdash} \alpha$, sino que necesariamente hemos de explicitar la deducción y escribir $\Gamma \overset{d}{\underset{F}{\vdash}} \alpha$. Aquí sobreentenderemos la d en lugar de escribirla explícitamente en todo momento.

Una fórmula α es un *teorema* de F si es la última fórmula de una demostración en F. Lo representaremos mediante $\underset{F}{\vdash} \alpha$.

Vamos a definir ahora un sistema deductivo formal (o, mejor dicho, uno para cada lenguaje formal) que incluya los axiomas y reglas de inferencia necesarios para capturar el concepto puro de razonamiento lógico tal y como lo entienden los matemáticos, pero sin introducir ninguna información sobre conjuntos, números naturales o cualquier otro concepto matemático concreto.

El sistema deductivo formal $K_{\mathcal{L}}$ Dado un lenguaje formal $\mathcal{L}$, llamaremos *axiomas de* $K_{\mathcal{L}}$ o *axiomas lógicos* a las fórmulas de $\mathcal{L}$ que sean de alguno de los tipos siguientes:[8]

Axiomas de $K_{\mathcal{L}}$

1. $\alpha \to (\beta \to \alpha)$,
2. $(\alpha \to (\beta \to \gamma)) \to ((\alpha \to \beta) \to (\alpha \to \gamma))$,
3. $(\neg\alpha \to \neg\beta) \to (\beta \to \alpha)$,
4. $\alpha \to \alpha \vee \beta, \quad \beta \to \alpha \vee \beta$,
5. $(\alpha \to \gamma) \to ((\beta \to \gamma) \to (\alpha \vee \beta \to \gamma))$,
6. $\alpha \to (\beta \to \alpha \wedge \beta)$,
7. $\alpha \wedge \beta \to \alpha, \quad \alpha \wedge \beta \to \beta$,
8. $(\alpha \to \beta) \to ((\beta \to \alpha) \to (\alpha \leftrightarrow \beta))$,
9. $(\alpha \leftrightarrow \beta) \to (\alpha \to \beta), \quad (\alpha \leftrightarrow \beta) \to (\beta \to \alpha)$,
10. $\bigwedge u\alpha \to \mathtt{S}^t_u\alpha$,
11. $\bigwedge u(\alpha \to \beta) \to (\alpha \to \bigwedge u\beta)$,
12. $\bigvee u\, \alpha \leftrightarrow \neg\bigwedge u\neg\alpha$.

    Sólo si $\mathcal{L}$ tiene igualador:

13. $\bigwedge u(u = t \to \alpha) \leftrightarrow \mathtt{S}^t_u\alpha$.

    Sólo si además $\mathcal{L}$ tiene descriptor:

14. $\overset{1}{\bigvee} u\alpha \to \mathtt{S}^{u|\alpha}_u\alpha$.
15. $\neg\overset{1}{\bigvee} u\alpha \to (u|\alpha) = v|(v = v)$.

[8] Es claro que en ARP podemos definir una fórmula $\alpha \in \mathrm{Ax}(\ulcorner K_{\mathcal{L}} \urcorner)$ que formalice la definición de axioma de $K_{\mathcal{L}}$. Hay un matiz entre la denominación "axioma lógico" y "axioma de $K_{\mathcal{L}}$" y es que es posible hacer otras elecciones de axiomas a los que es igualmente razonable llamar "axiomas lógicos", de modo que los axiomas de $K_{\mathcal{L}}$ representan una elección concreta de axiomas lógicos entre otras alternativas posibles.

Nota A partir de aquí entenderemos que todas las cadenas de signos que escribamos serán términos y fórmulas (nunca semitérminos o semifórmulas) salvo que se indique explícitamente. Por ejemplo, en el axioma 11 hay que entender que la variable ligada u no aparece libre en α y en el caso del axioma 13 es necesario que t sea un término y α una semifórmula en la que la única variable ligada que puede aparecer libre es u. ∎

En teoría podemos tomar las fórmulas que queramos como axiomas de un sistema deductivo formal, pero lo cierto es que la elección de los axiomas de $K_{\mathcal{L}}$ no es casual. Veremos que los axiomas 1, 2, 11 son suficientes para demostrar que $K_{\mathcal{L}}$ satisface el teorema de deducción 1.20, mientras que los axiomas 3–9 regulan formalmente el comportamiento de los conectores lógicos distintos del implicador (es decir, expresan que cada uno de ellos significa lo que queremos que signifique sin aludir a su significado). El axioma 10 expresa que $\bigwedge$ significa "para todo" y 12 determina el significado del particularizador. Finalmente, 13 determina el comportamiento del igualador y 14 y 15 el del descriptor.

El concepto de modelo nos permite probar que la elección de los axiomas y reglas de inferencia de $K_{\mathcal{L}}$ no es arbitraria en un sentido mucho más preciso que las observaciones precedentes:

Teorema 3.14 *Si M es un modelo de un lenguaje formal $\mathcal{L}$, entonces todos los axiomas de $K_{\mathcal{L}}$ son verdaderos en M.*

DEMOSTRACIÓN: La comprobación de los axiomas 1–9 es una aplicación rutinaria de la definición de satisfacción. Veamos, por ejemplo, el segundo:

Fijada una valoración v en M, tenemos que probar que si

$$M \vDash (\alpha \to (\beta \to \gamma))[v],$$

entonces también

$$M \vDash ((\alpha \to \beta) \to (\alpha \to \gamma))[v],$$

para lo cual, a su vez, hay que probar que si se cumple $M \vDash (\alpha \to \beta)[v]$, también $M \vDash (\alpha \to \gamma)[v]$, y a su vez tenemos que suponer que $M \vDash \alpha[v]$ y tenemos que probar que $M \vDash \gamma[v]$. En total, estamos suponiendo

$$M \vDash (\alpha \to (\beta \to \gamma))[v], \quad M \vDash (\alpha \to \beta)[v], \quad M \vDash \alpha[v].$$

Por la definición de satisfacción de una implicación de aquí se sigue que

$$M \vDash (\beta \to \gamma)[v], \quad M \vDash \beta[v],$$

y a su vez, de aquí se sigue que $M \vDash \gamma[v]$, como teníamos que probar.

El décimo axioma expresa que si $\bigwedge u\, \alpha$ es verdadero, también tiene que serlo cada fórmula $\mathsf{S}^t_u\alpha$. Para probarlo, fijada una valoración v en M, suponemos que $M \vDash \bigwedge u\alpha[v]$ y tenemos que probar que $M \vDash \mathsf{S}^t_u\alpha[v]$. Por el teorema 3.12, esto es lo mismo que $M \vDash \alpha[v_u^{M(t)[v]}]$. La conclusión es inmediata: por definición de satisfacción, tenemos que, para todo a en M, se cumple $M \vDash \alpha[v_u^a]$, y basta aplicar esto a $a = M(t)[v]$.

La idea subyacente a los axiomas de tipo 11 es que

Todo el mundo, si llueve, coge paraguas.

implica que

Si llueve, todo el mundo coge paraguas.

Una vez más, para probar que una implicación es satisfecha, tomamos como hipótesis que $M \vDash \bigwedge u(\alpha \to \beta)[v]$ (donde α es una fórmula y, en particular no tiene libre la variable u) y tenemos que probar $M \vDash (\alpha \to \bigwedge u\beta)[v]$, para lo cual, a su vez, podemos suponer $M \vDash \alpha[v]$, y tenemos que probar que $M \vDash \bigwedge u\,\beta[v]$.

Para ello fijamos un a en M y, por definición de satisfacción, sabemos que $M \vDash (\alpha \to \beta)[v_u^a]$. Además, $M \vDash \alpha[v_u^a]$, pues esto se cumple con la valoración v y, como u no está libre en α, sigue siendo cierto aunque modifiquemos la interpretación de u. Por lo tanto, por definición de la satisfacción de una implicación, tenemos que $M \vDash \beta[v_u^a]$ y, como esto vale para todo a de M, tenemos que $M \vDash \bigwedge u\beta[v]$.

La comprobación para los axiomas de tipo 12 es inmediata sin más que aplicar la definición de satisfacción.

A partir de aquí suponemos que $\mathcal{L}$ tiene igualador, con lo que sabemos que $M(=)$ se interpreta como la relación de identidad en M. Observamos que

$$M \vDash \bigwedge u(u = t \to \alpha)[v]$$

equivale a que, para todo a en M, se cumpla

$$M \vDash (u = t \to \alpha)[v_u^a],$$

lo que a su vez equivale a que si $a = M(t)[v]$, entonces $M \vDash \alpha[v_u^a]$. Así pues, $M \vDash \bigwedge u(u = t \to \alpha)[v]$ es equivalente a que $M \vDash \alpha[u^{M(t)[v]}]$, que, por el teorema 3.12, es equivalente a $M \vDash \mathtt{S}_u^t\alpha[v]$. Por definición de satisfacción de una coimplicación, esto equivale a que el axioma es satisfecho.

Si $\mathcal{L}$ tiene descriptor, $M \vDash (\overset{1}{\bigvee} u\alpha \to \mathtt{S}_u^{u|\alpha}\alpha)[v]$ equivale a que si existe un único a en M tal que $M[\alpha][v_u^a]$, entonces $M \vDash (\mathtt{S}_u^{u|\alpha}\alpha)[v]$, que a su vez equivale a que $M \vDash \alpha[v_u^{M(u|\alpha)[v]}]$, es decir, a que $M(u|\alpha)[v]$ sea precisamente el único a que cumple $M[\alpha][v_u^a]$, y esto es cierto por definición de $M(u|\alpha)[v]$. Por lo tanto, los axiomas de tipo 14 son verdaderos en M.

Finalmente, $M \vDash (\neg\overset{1}{\bigvee} u\alpha \to (u|\alpha) = w|(w = w))[v]$ equivale a que si no existe un único a en M tal que $M \vDash \alpha[v_u^a]$, entonces $M(u|\alpha)[v] = M(w|(w = w))[v]$. Por definición, en este caso $M(u|\alpha)[v]$ es la descripción impropia d fijada por el modelo. En cuanto a $M(w|(w = w))[v]$, pueden darse dos casos: si M tiene más de un elemento, entonces no existe un único a en M que cumpla $M \vDash (w = w)[v_u^a]$, ya que esto lo cumplen todos los objetos de M, luego $M(w|(w = w))[v]$ es la descripción impropia d. La otra posibilidad es que M sólo tenga un objeto, que necesariamente será d, luego también en este caso $M(w|(w = w))[v] = d$, y en ambos casos el axioma es satisfecho en M. ∎

El teorema anterior es un ejemplo típico de enunciado sobre "la totalidad de los modelos de un lenguaje formal" que tiene sentido a pesar de que no tenemos un concepto intuitivo de lo que es dicha totalidad porque tenemos un razonamiento para justificarla: sabemos que es cierto que cualquier axioma lógico será verdadero en cualquier modelo que consideremos porque tenemos un argumento que permite demostrar que lo es sea cual sea el modelo considerado.

Hemos definido los axiomas de $K_{\mathcal{L}}$, pero, para completar la definición de $K_{\mathcal{L}}$ como sistema deductivo formal, tenemos que fijar sus reglas de inferencia:

Definición 3.15 Dado un lenguaje formal $\mathcal{L}$, las reglas de inferencia primitivas de $K_{\mathcal{L}}$ son el *modus ponendo ponens* (MP) y la regla de *introducción del generalizador* o *regla de generalización:*

$$\text{(MP)}\ \frac{\alpha \qquad \alpha \to \beta}{\beta}, \qquad \text{(IG)}\ \frac{\alpha}{\bigwedge u\, \mathtt{S}_x^u \alpha},$$

donde, en IG, la variable x puede sustituirse por u en α en el sentido de la definición 3.11. Diremos que x es la *variable propia* de la regla IG.

La regla MP la conocemos ya, porque es válida en ARP, mientras que IG establece que las variables libres representan objetos arbitrarios exactamente igual que en ARP, de modo que toda variable libre puede ligarse con un generalizador.

Notemos que, dada una fórmula α, siempre podemos elegir una variable u tal que x pueda sustituirse por u en α. Basta con que u no esté en α.

Esto completa la definición de $K_{\mathcal{L}}$. En la práctica, escribiremos $\Gamma \vdash \alpha$ en lugar de $\Gamma \underset{K_{\mathcal{L}}}{\vdash} \alpha$, es decir, que cuando no indiquemos el sistema deductivo formal en el que se lleva a cabo una deducción, se deberá entender que es $K_{\mathcal{L}}$. Diremos además que α es una *consecuencia lógica* de Γ y a los teoremas de $K_{\mathcal{L}}$ los llamaremos *teoremas lógicos.*

Observemos que las dos reglas de inferencia de $K_{\mathcal{L}}$ son lógicamente válidas en el sentido de que, si sus premisas son verdaderas en un modelo de $\mathcal{L}$, la consecuencia también tiene que serlo.

En efecto, en el caso de MP es obvio, y el caso de IG también es sencillo: supongamos que $M \vDash \alpha$ y vamos a ver que también $M \vDash \bigwedge u \mathtt{S}_x^u \alpha$. Fijada una valoración v, tenemos que $M \vDash \alpha[v]$ y tenemos que probar que $M \vDash \bigwedge u\, S_x^u \alpha[v]$, lo cual, por 3.12, es equivalente a $M \vDash \alpha[v_x^{v(u)}]$, y esto es cierto por la definición de satisfacción de una generalización.

Como consecuencia:

Teorema 3.16 (Teorema de corrección) *Sea $\mathcal{L}$ un lenguaje formal y consideremos unas fórmulas $\gamma_1, \ldots, \gamma_n, \alpha$ de $\mathcal{L}$ tales que*

$$\gamma_1, \ldots, \gamma_n \vdash \alpha.$$

Si M es un modelo de $\mathcal{L}$ tal que $M \vDash \gamma_1, \ldots, \gamma_n$, entonces necesariamente se cumple también que $M \vDash \alpha$. En particular, todo teorema lógico es verdadero en cualquier modelo de $\mathcal{L}$.

En otras palabras: al razonar formalmente en $K_{\mathcal{L}}$ estamos razonando correctamente: aunque para razonar formalmente no es necesario suponer ningún significado en las fórmulas que forman parte de una deducción, lo cierto es que si aceptamos que significan algo (es decir, que hemos fijado un modelo del lenguaje formal considerado) y las premisas son verdaderas, las conclusiones que obtengamos formalmente también serán verdaderas.

DEMOSTRACIÓN: Sea $\alpha_1, \ldots, \alpha_m$ una deducción de $\alpha \equiv \alpha_m$ en $K_{\mathcal{L}}$. Basta razonar que cada α_i tiene que ser verdadera en M. En caso contrario, sea i el menor índice correspondiente a una fórmula que no sea verdadera en M. No puede ser que α_i sea una premisa, pues estamos suponiendo que todas ellas son verdaderas en M, ni tampoco puede ser un axioma lógico, ya que el teorema 3.14 prueba que todos ellos son verdaderos en M. Pero la única alternativa es que α_i sea consecuencia de fórmulas anteriores por una de las dos reglas de inferencia de $K_{\mathcal{L}}$, y dichas fórmulas anteriores serán verdaderas en M por la minimalidad de i, y antes del enunciado de este teorema hemos razonado que esto implica que α_i también tiene que serlo, y así tenemos una contradicción. ∎

Con esto hemos probado que las consecuencias lógicas que hemos definido formalmente son realmente consecuencias lógicas en el sentido informal de "hechos necesariamente ciertos si lo son las premisas". En 7.38 demostraremos el teorema de completitud semántica, que nos asegurará el recíproco: si una fórmula es necesariamente verdadera cuando lo son unas premisas dadas, entonces es consecuencia lógica en el sentido formal. La prueba del teorema de completitud se basa en razonamientos no triviales con modelos, así que la pospondremos hasta que hayamos formalizado la teoría de modelos. Hasta ese momento no vamos a demostrar más teoremas informalmente, y en particular no demostraremos más teoremas sobre modelos.

La única razón por la que hemos anticipado el concepto de "modelo" antes de estar en condiciones de formalizarlo es que así hemos demostrado que $K_{\mathcal{L}}$ es "fiable", en el sentido de que ahora sabemos que todo lo que demostremos en $K_{\mathcal{L}}$ a partir de unas premisas intuitivamente verdaderas es necesariamente intuitivamente verdadero, exactamente igual que si lo hubiéramos demostrado informalmente. De hecho, la prueba que hemos dado del teorema de corrección muestra explícitamente cómo convertir en un razonamiento informal concluyente cualquier deducción formal en $K_{\mathcal{L}}$.

A partir de la definición de $K_{\mathcal{L}}$ no es evidente en absoluto que cualquier razonamiento lógico informal que sea concluyente (en el sentido de que su conclusión sea necesariamente verdadera cuando lo son sus premisas) sea necesariamente formalizable en $K_{\mathcal{L}}$ (supuesto que las premisas y la conclusión del razonamiento sean formalizables en el lenguaje formal $\mathcal{L}$). Esto nos lo asegurará el teorema de completitud semántica, que demostraremos más adelante, pero de momento vamos a probar algunos resultados que basten en la práctica para que, con un poco de práctica, formalizar en $K_{\mathcal{L}}$ un argumento informal dado no ofrezca ninguna dificultad. De momento, la situación es que no es evidente en absoluto cómo formalizar en $K_{\mathcal{L}}$ ni siquiera los argumentos lógicos más elementales.

El resultado fundamental es un teorema de deducción análogo al que hemos demostrado para ARP. Para probarlo demostramos en primer lugar una consecuencia de los axiomas de tipos 1 y 2:

$$\vdash \alpha \to \alpha.$$

En efecto, la primera línea de la demostración siguiente es un axioma de tipo 2 tomando como fórmulas α, β, γ las fórmulas $\alpha, \alpha \to \alpha, \alpha$, respectivamente:

(1)	$(\alpha \to ((\alpha \to \alpha) \to \alpha)) \to ((\alpha \to (\alpha \to \alpha)) \to (\alpha \to \alpha))$	Ax. 2
(2)	$\alpha \to ((\alpha \to \alpha) \to \alpha)$	Ax. 1
(3)	$(\alpha \to (\alpha \to \alpha)) \to (\alpha \to \alpha)$	MP 1, 2
(4)	$\alpha \to (\alpha \to \alpha)$	Ax 1
(5)	$\alpha \to \alpha$	MP 3, 4

Ahora ya podemos probar:

Teorema 3.17 (Teorema de deducción) *Fijado un lenguaje formal $\mathcal{L}$, si se cumple*

$$\alpha_1, \ldots, \alpha_n, \alpha \vdash \beta$$

y en la deducción no se usa la regla IG *con variables propias libres en α, entonces*

$$\alpha_1, \ldots, \alpha_n \vdash \alpha \to \beta.$$

Además, en esta deducción se usa IG *exactamente respecto de las mismas variables que en la dada.*

En la prueba que damos a continuación conviene observar que —como habíamos anticipado— para que se cumpla el teorema de deducción basta con que entre los axiomas lógicos se encuentren los de tipo 1, 2 y 11, es decir, que el teorema se seguiría cumpliendo aunque no incluyéramos los demás y también si incluyéramos otras fórmulas cualesquiera como axiomas adicionales.

DEMOSTRACIÓN: Por hipótesis existe una deducción $\delta_1, \ldots, \delta_m$ con premisas en Γ y α tal que $\delta_m \equiv \beta$ y en la que no se generaliza con variables propias libres en α.

Vamos a construir una deducción con premisas en Γ que contenga las fórmulas $\alpha \to \delta_i$ (con otras posibles fórmulas intercaladas). Como la última de estas fórmulas es $\alpha \to \beta$, con esto tendremos que la implicación es consecuencia de Γ.

Si δ_i es un axioma lógico o una premisa distinta de α, la forma de incorporar $\alpha \to \delta_i$ a la deducción es la siguiente:

(1)	δ_i	axioma o premisa
(2)	$\delta_i \to (\alpha \to \delta_i)$	Axioma 1
(3)	$\alpha \to \delta_i$	MP 1, 2

Si $\delta_i \equiv \alpha$, entonces debemos incorporar a la deducción que estamos construyendo la fórmula $\alpha \to \alpha$, y ya hemos observado que esto es un teorema lógico (que se demuestra sin usar IG).

Si δ_i se deduce por MP, entonces hay dos fórmulas anteriores de la forma δ_j y $\delta_j \to \delta_i$. Puesto que en nuestra deducción vamos incorporando las implicaciones de forma sucesiva, cuando lleguemos a δ_i ya habremos incorporado las fórmulas $\alpha \to \delta_j$ y $\alpha \to (\delta_j \to \delta_i)$. Éstas son, pues, líneas anteriores de nuestra deducción y podemos usarlas para deducir $\alpha \to \delta_i$.

Lo hacemos de este modo:

(1)	$\alpha \to \delta_j$	fórmula anterior
(2)	$\alpha \to (\delta_j \to \delta_i)$	fórmula anterior
(3)	$(\alpha \to (\delta_j \to \delta_i)) \to ((\alpha \to \delta_j) \to (\alpha \to \delta_i))$	Axioma 2
(4)	$(\alpha \to \delta_j) \to (\alpha \to \delta_i)$	MP 2, 3
(5)	$\alpha \to \delta_i$	MP 1, 4

Supongamos, por último que δ_i se deduce por generalización de otra fórmula anterior δ_j. Esto significa que $\delta_i \equiv \bigwedge u\ \mathtt{S}^u_x\delta_j$, donde la variable x puede sustituirse por u en δ_j y, por la hipótesis del teorema, sabemos que x no está libre en α. Entonces incorporamos $\alpha \to \bigwedge u\ \mathtt{S}^u_x\delta_j$ de este modo:

(1)	$\alpha \to \delta_j$	fórmula anterior
(2)	$\bigwedge u(\alpha \to \mathtt{S}^u_x\delta_j)$	IG 1
(3)	$\bigwedge u(\alpha \to \mathtt{S}^u_x\ \delta_j) \to (\alpha \to \bigwedge u\ S^u_x\delta_j)$	Axioma 11
(4)	$\alpha \to \bigwedge u\ \mathtt{S}^u_x\delta_j$	MP 2, 3

Notemos que, como x no está libre en α, puede sustituirse por u en α (y la sustitución deja a α inalterada), por lo que también es posible sustituir x por u en $\alpha \to \delta_j$ y, por lo tanto, el uso de IG en (2) es correcto. El hecho de que u no esté libre en α (porque α es una fórmula) justifica también que la línea (3) es un axioma. Esto completa la prueba.[9] ∎

En la práctica podemos usar el teorema de deducción en una versión ligeramente más general, análogamente a como hemos hecho en el caso de ARP, y que describimos aquí de nuevo por completitud:

Imaginemos que estamos construyendo una deducción, digamos $\delta_1, \ldots, \delta_m$, a partir de unas premisas Γ, y a continuación queremos deducir una fórmula de tipo $\alpha \to \beta$. Entonces escribimos α y la usamos como una premisa más hasta obtener β (sin generalizar respecto de variables propias libres en α). Al llegar a β, lo que hemos probado es que $\delta_1, \ldots, \delta_m, \alpha \vdash \beta$, con una deducción en la que no se generaliza respecto de variables propias libres en α, luego el teorema de deducción nos da que $\delta_1, \ldots, \delta_m \vdash \alpha \to \beta$, es decir, que existe una deducción de $\alpha \to \beta$ con premisas en $\delta_1, \ldots, \delta_m$. Pero por otra parte sabemos que $\delta_1, \ldots, \delta_m$ se deducen de Γ, luego también $\Gamma \vdash \alpha \to \beta$. (Para tener una deducción que prueba esto deducimos cada δ_i de Γ y luego deducimos $\alpha \to \beta$ de las δ_i).

[9] Técnicamente, la prueba que hemos dado consiste en definir un funtor $\mathrm{td}(\Gamma, d, \alpha, \beta)$, de modo que

$$\Gamma \cup \{\alpha\} \overset{d}{\vdash} \beta \to \Gamma \overset{\mathrm{td}(\Gamma,d,\alpha,\beta)}{\vdash} \alpha \ulcorner\to\urcorner \beta.$$

En la práctica seguiremos la costumbre que ya aplicábamos en ARP de marcar todas las líneas desde que suponemos α hasta que llegamos a β con una línea vertical. Esta línea advierte de que las fórmulas abarcadas por ella no son consecuencia de las premisas de la deducción principal, sino de las premisas más una hipótesis auxiliar (la fórmula α, que marcaremos con la etiqueta de "hipótesis") que sólo hemos aceptado provisionalmente para aplicar el teorema de deducción. Si una vez hemos añadido $\alpha \to \beta$ a la deducción prosiguiéramos haciendo uso de las líneas marcadas, la deducción sería inválida, pues estaríamos haciendo uso de una hipótesis local α que no forma parte de las premisas de la deducción.

La observación de que para obtener la deducción cuya existencia afirma el teorema de deducción sólo se generaliza respecto de variables propias respecto a las que ya se generalizaba en la deducción dada es esencial para que estemos seguros de que no estamos generalizando en un momento dado respecto de una variable "prohibida". El uso del teorema de deducción oculta el uso de algunos axiomas y reglas de inferencia (las que aparecen en la demostración al construir la deducción de $\alpha \to \beta$), pero no oculta ningún uso de IG.

Un uso incorrecto del teorema de deducción Consideramos el lenguaje formal $\mathcal{L}^+_{\mathrm{arp}}$:

(1)	$x = 0$	Hipótesis
(2)	$\bigwedge u\ u = 0$	IG 1
(3)	$x = 0 \to \bigwedge u\ u = 0$	TD (incorrecto)
(4)	$\bigwedge v(v = 0 \to \bigwedge u\ u = 0)$	IG 3
(5)	$\bigwedge v(v = 0 \to \bigwedge u\ u = 0) \to S^0_v(v = 0 \to \bigwedge u\ u = 0)$	Axioma 4
(6)	$0 = 0 \to \bigwedge u\ u = 0$	MP 4,5
(7)	$0 = 0$	Teorema lógico
(8)	$\bigwedge u\ u = 0$	MP 6,7

La línea (3) no está justificada porque en (2) hemos generalizado respecto a la variable propia x, que está libre en la hipótesis. Veremos un poco más adelante que la línea (7) es un teorema lógico de cualquier lenguaje con igualador. En realidad, podemos razonar directamente que (3) no puede ser un teorema lógico, pues es una fórmula falsa en cualquier modelo cuyo universo tenga al menos dos elementos, pero es más evidente en el caso de la línea (8).[10]

La interpretación de fondo es la misma que en el caso de ARP: en principio, una variable libre representa un objeto arbitrario, pero cuando tomamos como hipótesis $x = 0$, ya no es posible entender que x se refiere a un objeto arbitrario, sino que x es concretamente $x = 0$, por lo que pasar a $\bigwedge uu = 0$ es incorrecto (no es necesariamente cierto que todo sea igual a 0). ∎

[10] En general, el lector podría haber pasado por alto todas las consideraciones semánticas sin que ello le impidiera seguir la exposición de la parte sintáctica, pero no podemos demostrar que (3) u (8) no son teoremas lógicos sin recurrir a argumentos semánticos. No obstante, el lector puede saltarse este ejemplo y limitarse a observar que sólo tenemos demostrado el teorema de deducción con la condición de no generalizar respecto de variables libres en la hipótesis.

Observemos que el recíproco del teorema de deducción es trivial:

Teorema 3.18 *Si $\alpha_1, \ldots, \alpha_n \vdash \alpha \to \beta$, entonces $\alpha_1, \ldots, \alpha_n, \alpha \vdash \beta$.*

Basta prolongar la deducción dada añadiendo α como premisa y luego aplicar MP para obtener β.

Veamos ahora que en $K_{\mathcal{L}}$ también es válida la demostración por reducción al absurdo. Para ello demostramos la *regla de inferencia de contradicción:*

$$\text{(C)} \quad \alpha, \neg\alpha \vdash \beta.$$

En efecto:

	(1)	$\neg\alpha$	Premisa
|	(2)	$\neg\beta$	Hipótesis
	(3)	$\neg\beta \to \neg\alpha$	
	(4)	$(\neg\beta \to \neg\alpha) \to (\alpha \to \beta)$	Axioma 3
	(5)	$\alpha \to \beta$	MP 3, 4
	(6)	α	Premisa
	(7)	β	MP 5, 6

La regla de contradicción expresa un principio lógico fundamental, y es que a partir de una contradicción α y $\neg\alpha$ se puede deducir cualquier cosa. Así ya podemos demostrar:

Teorema 3.19 (Reducción al absurdo) *Si en una deducción tomamos $\neg\alpha$ como premisa adicional y, sin generalizar respecto de variables libres en α, llegamos a deducir una fórmula β y también $\neg\beta$, podemos concluir α.*

Más precisamente, lo que afirma el teorema es que en esta situación:

(1)	γ_1	
⋮		deducción a partir de unas premisas $\alpha_1, \ldots, \alpha_n$
(m)	γ_m	
$(m+1)$	$\neg\alpha$	Hipótesis
⋮		deducción a partir de las premisas, las líneas precedentes y $\neg\alpha$
(k)	β	
⋮		
(l)	$\neg\beta$	
$(l+1)$	α	Reducción al absurdo
⋮		

si a partir del momento en que suponemos $\neg\alpha$ no generalizamos respecto a variables propias libres en α, la fórmula α escrita en la línea $l+1$ es deducible a partir de las premisas, pero las líneas marcadas con la raya vertical a la izquierda no lo son, porque suponen la hipótesis adicional $\neg\alpha$.

DEMOSTRACIÓN: La regla de inferencia (C) nos permite añadir tras la línea (l) la línea $\neg(\alpha \to \alpha)$, y el teorema de deducción nos da que $\neg\alpha \to \neg(\alpha \to \alpha)$ es deducible de las premisas. El axioma 3, junto con MP, nos permite prolongar la deducción con $(\alpha \to \alpha) \to \alpha$, pero $\alpha \to \alpha$ es un teorema lógico, luego podemos incluirlo y llegar a α por MP. En la práctica suprimiremos estas líneas y escribiremos directamente α tras haber llegado a una contradicción. Igualmente, usando la regla de la doble negación (que probamos a continuación), vemos que si suponemos α y llegamos a una contradicción podemos concluir $\neg\alpha$. ∎

3.4 Reglas derivadas de inferencia

El paso siguiente para que $K_{\mathcal{L}}$ sea "manejable en la práctica" es demostrar reglas de inferencia derivadas, es decir, resultados del tipo

$$\alpha_1, \ldots, \alpha_n \vdash \alpha,$$

como es el caso de la regla de contradicción (C) que hemos probado antes del teorema 3.19, que podemos usar en la práctica incluyendo directamente α en cualquier deducción que contenga ya las premisas $\alpha_1, \ldots, \alpha_n$, sabiendo que el "agujero" que queda se puede "rellenar" insertando la deducción de la regla.

Hay una regla trivial que conviene discutir:

Regla de repetición (R) $\alpha \vdash \alpha$.

Esto es trivial, por la propia definición de deducción, pero no es igual de obvio que en una deducción podemos repetir una fórmula precedente. Ahora bien, una forma de hacerlo es demostrar $\alpha \to \alpha$ y luego aplicar MP, con lo que volvemos a tener α.

En teoría nunca es necesario usar esta regla, pues si una fórmula está en una deducción podemos usarla siempre que queramos sin necesidad de repetirla, pero a veces es útil escribir una misma fórmula con una notación distinta, como pasar de $\mathtt{S}^t_x(x = x)$ a $t = t$.

3.4.1 Reglas relacionadas con los conectores lógicos

Empezamos demostrando la regla cuya deducción hemos dejado pendiente en la prueba del teorema 3.19:

Reglas de la doble negación (DN) $\neg\neg\alpha \vdash \alpha, \qquad \alpha \vdash \neg\neg\alpha$.

DEMOSTRACIÓN: Si suponemos $\neg\neg\alpha$, suponemos $\neg\alpha$ por reducción al absurdo (por la variante que ya hemos demostrado) y así tenemos la contradicción $\neg\alpha$ y $\neg\neg\alpha$, luego podemos concluir α. Esto prueba la primera regla, y el teorema de deducción nos da que $\vdash \neg\neg\alpha \to \alpha$, pero esto vale para toda fórmula α, luego en particular, para $\neg\alpha$, luego tenemos que $\vdash \neg\neg\neg\alpha \to \neg\alpha$, y el axioma 3 nos da entonces $\vdash \alpha \to \neg\neg\alpha$, de donde MP nos da la segunda regla. ∎

Aplicando MP a los axiomas de tipo 4 y 5 de $K_{\mathcal{L}}$ obtenemos inmediatamente las reglas fundamentales del disyuntor:

Reglas de introducción del disyuntor y del dilema

$$\text{(ID)}\quad \alpha \vdash \alpha \vee \beta, \quad \beta \vdash \alpha \vee \beta \qquad \text{(Dil)}\quad \alpha \to \gamma, \beta \to \gamma \vdash \alpha \vee \beta \to \gamma.$$

Con ellas podemos probar:

Regla del Tertium non datur (TND) $\vdash \alpha \vee \neg\alpha$.

Demostración: Razonamos por reducción al absurdo:

⎸	(1)	$\neg(\alpha \vee \neg\alpha)$	Hipótesis
⎸⎸	(2)	$\neg\alpha$	Hipótesis
⎸⎸	(3)	$\alpha \vee \neg\alpha$	ID 2
⎸	(4)	α	RA 1, 3
⎸	(5)	$\alpha \vee \neg\alpha$	ID 4
	(6)	$\alpha \vee \neg\alpha$	RA 1, 5

Aplicando MP a los axiomas de tipo 6 y 7 de $K_{\mathcal{L}}$ obtenemos:

Reglas de introducción y eliminación del conjuntor

$$\text{(IC)}\quad \alpha, \beta \vdash \alpha \wedge \beta, \qquad \text{(EC)}\quad \alpha \wedge \beta \vdash \alpha, \quad \alpha \wedge \beta \vdash \beta.$$

Igualmente, con los axiomas de tipo 8 y 9 obtenemos:

Reglas de introducción y eliminación del bicondicionador

$$\text{(IB)}\quad \alpha \to \beta, \beta \to \alpha \vdash \alpha \leftrightarrow \beta, \qquad \text{(EB)}\quad \alpha \leftrightarrow \beta \vdash \alpha \to \beta, \quad \alpha \leftrightarrow \beta \vdash \beta \to \alpha.$$

Estas reglas de inferencia, junto con el teorema de deducción y el teorema sobre la reducción al absurdo, equivalen a los axiomas 1–9, en el sentido de que su uso hace innecesario usar directamente dichos axiomas, que en algunos casos son mucho más artificiosos. Esto se pone de manifiesto en las demostraciones de las reglas de inferencia siguientes:

Reglas de equivalencia entre el disyuntor y el implicador (EDI):

$$\alpha \vee \beta \vdash \neg\alpha \to \beta, \qquad \neg\alpha \to \beta \vdash \alpha \vee \beta$$

$$\alpha \to \beta \vdash \neg\alpha \vee \beta, \quad \neg\alpha \vee \beta \vdash \alpha \to \beta.$$

DEMOSTRACIÓN: Probamos las dos primeras formas, pues las siguientes se prueban del mismo modo:

(1)	$\alpha \vee \beta$	Premisa
(2)	$\neg\alpha$	Hipótesis
(3)	α	Hipótesis
(4)	β	C 2, 3
(5)	$\alpha \to \beta$	
(6)	$\beta \to \beta$	Teorema lógico
(7)	$\alpha \vee \beta \to \beta$	Dil 5, 6
(8)	β	MP 1, 7
(9)	$\neg\alpha \to \beta$	

(1)	$\neg\alpha \to \beta$	Premisa
(2)	$\neg\alpha$	Hipótesis
(3)	β	MP 1, 2
(4)	$\alpha \vee \beta$	ID 3
(5)	$\neg\alpha \to \alpha \vee \beta$	
(6)	$\alpha \to \alpha \vee \beta$	ID
(7)	$\alpha \vee \neg\alpha \to \alpha \vee \beta$	Dil 5, 6
(8)	$\alpha \vee \neg\alpha$	TND
(9)	$\alpha \vee \beta$	MP 7, 8

Leyes de De Morgan (DM):

$$\alpha \wedge \beta \vdash \neg(\neg\alpha \vee \neg\beta), \quad \neg(\neg\alpha \vee \neg\beta) \vdash \alpha \wedge \beta,$$

$$\neg(\alpha \wedge \beta) \vdash \neg\alpha \vee \neg\beta, \quad \neg\alpha \vee \neg\beta \vdash \neg(\alpha \wedge \beta),$$

$$\alpha \vee \beta \vdash \neg(\neg\alpha \wedge \neg\beta), \quad \neg(\neg\alpha \wedge \neg\beta) \vdash \alpha \vee \beta,$$

$$\neg(\alpha \vee \beta) \vdash \neg\alpha \wedge \neg\beta, \quad \neg\alpha \wedge \neg\beta \vdash \neg(\alpha \vee \beta).$$

DEMOSTRACIÓN: Probamos, por ejemplo, las dos primeras formas:

(1)	$\alpha \wedge \beta$	Premisa
(2)	α	EC 1
(3)	β	EC 1
(4)	$\neg\alpha \vee \neg\beta$	Hipótesis
(5)	$\neg\neg\alpha \to \neg\beta$	EDI 4
(6)	$\neg\neg\alpha$	DN 2
(7)	$\neg\beta$	MP 5, 6
(8)	$\neg(\neg\alpha \vee \neg\beta)$	RA 3, 7

(1)	$\neg(\neg\alpha \vee \neg\beta)$	Premisa
(2)	$\neg\alpha$	Hipótesis
(3)	$\neg\alpha \vee \neg\beta$	ID 2
(4)	α	RA 1, 3
(5)	$\neg\beta$	Hipótesis
(6)	$\neg\alpha \vee \neg\beta$	ID 5
(7)	β	RA 1, 6
(8)	$\alpha \wedge \beta$	IC 4, 7

Definición de conectores De las reglas IB, EB, IC, EC se sigue inmediatamente que

$$\alpha \leftrightarrow \beta \vdash (\alpha \to \beta) \wedge (\beta \to \alpha), \qquad (\alpha \to \beta) \wedge (\beta \to \alpha) \vdash \alpha \leftrightarrow \beta.$$

De aquí se desprende que podemos modificar la definición de lenguaje formal eliminando el coimplicador y, consecuentemente, suprimiendo los axiomas 8 y 9 de la definición de $K_{\mathcal{L}}$. Si en estas condiciones definimos

$$\alpha \leftrightarrow \beta \equiv (\alpha \to \beta) \wedge (\beta \to \alpha),$$

obtenemos un cálculo deductivo equivalente en el sentido de que, por una parte, toda deducción en $K_{\mathcal{L}}$ con el coimplicador como conector primitivo se traduce en una deducción con las mismas premisas y la misma conclusión en la versión de $K_{\mathcal{L}}$ con el coimplicador como conector definido, pues los axiomas 8 y 9 se

demuestran fácilmente y, recíprocamente, toda deducción en la versión con el coimplicador definido se traduce en una deducción en la versión con el coimplicador primitivo porque aprovechar que $\alpha \leftrightarrow \beta$ es lo mismo que $(\alpha \rightarrow \beta) \wedge (\beta \rightarrow \alpha)$ se traduce en una aplicación de las reglas de inferencia que acabamos de enunciar. Así pues:

> *En lo sucesivo consideraremos lenguajes formales sin coimplicador, en los que éste se define como hemos indicado (y en $K_{\mathcal{L}}$ eliminamos los axiomas de tipo 8 y 9).*

Las leyes de De Morgan nos permiten hacer lo mismo con el conjuntor, es decir, podemos eliminar el conjuntor de la definición de lenguaje formal (y los axiomas de tipo 6 y 7 de la definición de $K_{\mathcal{L}}$) y definir

$$\alpha \wedge \beta \equiv \neg(\neg\alpha \vee \neg\beta).$$

En esta versión de $K_{\mathcal{L}}$ podemos demostrar las reglas IC, EC, y a partir de ellas, trivialmente, lo que eran los axiomas 6 y 7. En efecto:

(1)	α	Premisa
(2)	β	Premisa
(3)	$\neg(\alpha \wedge \beta)$	Hipótesis
(4)	$\neg\neg(\neg\alpha \vee \neg q)$	R 3
(5)	$\neg\alpha \vee \neg\beta$	DN 4
(6)	$\neg\neg\alpha \rightarrow \neg\beta$	EDI 5
(7)	$\neg\neg\alpha$	DN 1
(8)	$\neg\beta$	MP 6, 7
(9)	$\alpha \wedge \beta$	RA 2, 8

(1)	$\alpha \wedge \beta$	Premisa
(2)	$\neg\alpha$	Hipótesis
(3)	$\neg\alpha \vee \neg\beta$	ID 2
(4)	$\neg(\neg\alpha \vee \neg\beta)$	R 1
(5)	α	RA 3, 4

Análogamente se deduce β.

Esto se traduce en que, como en el caso del bicondicionador, cualquier deducción en la versión de $K_{\mathcal{L}}$ con el conjuntor como signo primitivo se traduce a otra en la versión con el conjuntos como signo definido y viceversa. Por ello:

> *En lo sucesivo consideraremos lenguajes formales sin conjuntor, en los que éste se define como hemos indicado (y en $K_{\mathcal{L}}$ eliminamos los axiomas de tipo 6 y 7).*

A su vez, las reglas de Equivalencia entre el disyuntor y el implicador permiten definir el disyuntor en términos del implicador, pero también el implicador en términos del disyuntor, y sucede que las dos posibilidades (que son equivalentes en la práctica) tienen interés.

Una de ellas consiste en eliminar el disyuntor de la lista de conectores primitivos (con lo que éstos se reducen a $\neg$ y $\rightarrow$) y definir

$$\alpha \vee \beta \equiv \neg\alpha \rightarrow \beta.$$

A su vez, suprimimos los axiomas de tipo 4 y 5 en la definición de $K_{\mathcal{L}}$, con lo que de los nueve primeros axiomas, ahora conservamos únicamente los tres primeros.

En esta versión con el disyuntor definido podemos demostrar las reglas ID y Dil (y a partir de ellas los axiomas de tipo 4 y 5). En efecto:

(1)	α	Premisa
(2)	$\neg\alpha$	Hipótesis
(3)	β	C 1, 2
(4)	$\neg\alpha \to \beta$	
(5)	$\alpha \vee \beta$	R 3

(1)	β	Premisa
(2)	$\neg\alpha$	Hipótesis
(3)	$\neg\alpha \to \beta$	
(4)	$\alpha \vee \beta$	R 3

(1)	$\alpha \to \gamma$	Premisa
(2)	$\beta \to \gamma$	Premisa
(3)	$\neg\alpha \to \beta$	Hipótesis ($\alpha \vee \beta$)
(4)	$\neg\gamma$	Hipótesis
(5)	α	Hipótesis
(6)	γ	MP 1, 5
(7)	$\neg\alpha$	RA 4, 6
(8)	β	MP 3, 7
(9)	γ	MP 2, 8
(10)	γ	RA4, 9
(11)	$\alpha \vee \beta \to \gamma$	

Esto hace que toda deducción en $K_{\mathcal{L}}$ con el disyuntor como conector primitivo se traduce en otra en $K_{\mathcal{L}}$ con el disyuntor definido y viceversa.

La alternativa consiste en considerar el implicador como definido por

$$\alpha \to \beta \equiv \neg\alpha \vee \beta$$

y mantener en $K_{\mathcal{L}}$ los axiomas 1–5. En este caso no hay que demostrar ningún axioma porque no hemos eliminado ninguno, y también es claro que toda deducción en $K_{\mathcal{L}}$ con el implicador primitivo se traduce en una deducción en $K_{\mathcal{L}}$ con el implicador definido y viceversa.

En la práctica no tiene ninguna relevancia qué conectores consideramos como primitivos (es decir, incluidos en la definición de lenguaje formal) y cuáles consideramos como definidos. Las fórmulas que podemos deducir son las mismas, aunque haya que interpretarlas como cadenas de signos distintas según cuáles sean los conectores primitivos, pero desde un punto de vista teórico ambas alternativas (eliminar el implicador o el disyuntor) tienen interés. Al considerar que los conectores primitivos son $\neg, \to$ los nueve primeros axiomas de $K_{\mathcal{L}}$ se reducen a los tres primeros, pero considerar $\neg, \vee$ como conectores primitivos es más conveniente para relacionar el cálculo deductivo que estamos estudiando aquí con el cálculo secuencial que estudiaremos en [CS]. Por ello:

> *En lo sucesivo consideraremos lenguajes formales cuyos únicos conectores primitivos sean* $\neg, \vee$.

No obstante, todo cuanto digamos se traduce trivialmente al caso de lenguajes formales con conectores primitivos $\neg, \to$ o también con los cinco conectores primitivos.

3.4.2 Reglas relacionadas con los cuantificadores

Los axiomas de tipo 10 nos dan inmediatamente:

Regla de eliminación del generalizador (EG): $\bigwedge u\alpha \vdash \mathtt{S}^t_u\alpha$.

Observemos que la prueba de esta regla se reduce a usar MP y un axioma de tipo 10, de modo que en ella no se usa IG, así que puede usarse libremente incluso en contextos en los que no esté permitido generalizar respecto de ciertas variables.

Reglas de negación del generalizador (NG):

$$\begin{array}{ll} \neg\bigwedge u\neg\alpha \vdash \bigvee u\alpha & \bigvee u\alpha \vdash \neg\bigwedge u\neg\alpha \\ \neg\bigwedge u\alpha \vdash \bigvee u\neg\alpha & \bigvee u\neg\alpha \vdash \neg\bigwedge u\alpha \end{array}$$

Demostración: Las dos primeras se siguen inmediatamente de los axiomas de tipo 12, y esto hace que podamos suprimir el particularizador de la definición de lenguaje formal, junto con los axiomas de tipo 12 de la definición de los axiomas de $K_{\mathcal{L}}$, y definir

$$\bigvee u\alpha \equiv \neg\bigwedge\neg\alpha.$$

Entonces las dos primeras reglas de NG son casos particulares de R, y permiten demostrar los axiomas de tipo 12. Veamos ahora la prueba de las dos últimas.

(1)	$\neg\bigwedge u\alpha$	Premisa	
(2)	$\neg\bigvee u\neg\alpha$	Hipótesis	
(3)	$\neg\bigwedge u\neg\neg\alpha$	Hipótesis	
(4)	$\bigvee u\neg\alpha$	NG (ya probada)	
(5)	$\bigwedge u\neg\neg\alpha$	RA 2, 4	
(6)	$\neg\neg\mathtt{S}^x_u\alpha$	EG 5	
(7)	$\mathtt{S}^x_u\alpha$	DN 6	
(8)	$\bigwedge u\mathtt{S}^u_x\mathtt{S}^x_u\alpha$	IG 7	
(9)	$\bigwedge u\alpha$	R 8	
(10)	$\bigvee u\neg\alpha$	RA 1, 9	

(1)	$\bigvee u\neg\alpha$	Premisa	
(2)	$\neg\bigwedge u\neg\neg\alpha$	NG (ya probada)	
(3)	$\bigwedge u\alpha$	Hipótesis	
(4)	$\mathtt{S}^x_u\alpha$	EG 3	
(5)	$\mathtt{S}^x_u\neg\neg\alpha$	DN 4	
(6)	$\bigwedge u\mathtt{S}^u_x\mathtt{S}^x_u\neg\neg\alpha$	IG 5	
(7)	$\bigwedge u\neg\neg\alpha$	R 6	
(8)	$\neg\bigwedge u\alpha$	RA 2, 7	

En ambas deducciones, x es una variable libre que no esté en α, de modo que en ambas es lícito el uso de IG, y en ambas usamos que, como es fácil probar (por inducción sobre la longitud de α), si x no está en la semifórmula α, entonces x se puede sustituir por u en $\mathtt{S}^x_u\alpha$ y $\mathtt{S}^u_x\mathtt{S}^x_u\alpha \equiv \alpha$.

Reglas de negación del particularizador (NP):

$$\begin{array}{ll} \neg\bigvee u\alpha \vdash \bigwedge u\neg\alpha & \bigwedge u\neg\alpha \vdash \neg\bigvee u\alpha \\ \neg\bigvee u\neg\alpha \vdash \bigwedge u\alpha & \bigwedge u\alpha \vdash \neg\bigvee u\neg\alpha \end{array}$$

Demostración: Se deducen trivialmente de las anteriores por reducción al absurdo. Por ejemplo,

(1)	$\neg\bigvee u\alpha$	Premisa	
(2)	$\neg\bigwedge u\neg\alpha$	Hipótesis	
(3)	$\bigvee u\alpha$	NG 2	
(4)	$\bigwedge u\neg\alpha$	RA 1, 2	

(1)	$\bigwedge u\neg\alpha$	Premisa	
(2)	$\bigvee u\alpha$	Hipótesis	
(3)	$\neg\bigwedge u\neg\alpha$	NG 2	
(4)	$\neg\bigvee u\alpha$	RA 1, 2	

Regla de introducción del particularizador (IP): $\mathtt{S}^t_u\alpha \vdash \bigvee u\alpha$.

Esta regla afirma que si hemos probado que un cierto término t cumple lo que dice α, entonces podemos afirmar que existe un u que cumple α.

Demostración:

(1)	$\mathtt{S}^t_u\alpha$	Premisa
(2)	$\neg\bigvee u\alpha$	Hipótesis
(3)	$\bigwedge u\neg\alpha$	NP 2
(4)	$\neg\mathtt{S}^t_u\alpha$	EG 3
(5)	$\bigvee u\alpha$	RA 1,4

Regla de eliminación del particularizador (EP) Este resultado no es realmente una regla de inferencia, aunque en la práctica lo podemos usar como tal. En realidad es un metateorema que nos garantiza la existencia de una deducción a partir de otra, como el teorema de deducción o el teorema sobre la reducción al absurdo. Concretamente, en esta situación:

(1)	γ_1	
⋮		deducción a partir de unas premisas $\alpha_1, \ldots, \alpha_n$
(k)	$\bigvee u\alpha$	
⋮		
(m)	γ_m	
$(m+1)$	$\mathtt{S}^x_u\alpha$	EP k
⋮		deducción a partir de las premisas, las líneas precedentes y $\mathtt{S}^x_u\alpha$

si la variable propia x no está en α y a partir de la línea $m+1$ no se generaliza respecto de variables libres en $\mathtt{S}^x_u\alpha$, entonces toda línea posterior β que no tenga libre la variable x es una consecuencia de las premisas.

Demostración: En principio, en la situación descrita tenemos que

$$\alpha_1, \ldots, \alpha_n, \mathtt{S}^x_u\alpha \vdash \beta.$$

Ahora bien, como para obtener β no se ha generalizado respecto de ninguna variable libre en $\mathtt{S}^x_u\alpha$, podemos aplicar el teorema de deducción y concluir que

$$\alpha_1, \ldots, \alpha_n \vdash \mathtt{S}^x_u\alpha \to \beta.$$

Pero también sabemos que $\alpha_1, \ldots, \alpha_n \vdash \bigvee u\alpha$, luego sólo necesitamos probar que $\mathtt{S}^x_u\alpha \to \beta, \bigvee u\alpha \vdash \beta$. En efecto:

(1)	$\mathtt{S}^x_u\alpha \to \beta$	Premisa
(2)	$\bigvee u\alpha$	Premisa
(3)	$\neg\bigwedge u\neg\alpha$	NG 2
(4)	$\neg\beta$	Hipótesis
(5)	$\mathtt{S}^x_u\alpha$	Hipótesis
(6)	β	MP 1, 5
(7)	$\neg\mathtt{S}^x_u\alpha$	RA 4,6
(8)	$\bigwedge u\neg\alpha$	IG 7
(9)	β	RA 3, 8

■

Es muy importante observar que en la deducción de β a partir de las premisas se generaliza respecto de la variable propia x. Esto significa que si aplicamos EP en un contexto en el que tenemos prohibido generalizar respecto de ciertas variables, debemos elegir la variable propia x como una nueva variable sobre la que no exista prohibición de generalizar.

Como al eliminar un particularizador dejamos libre una variable que no puede quedar libre en la conclusión, una forma de volver a ligarla es mediante la regla de introducción del particularizador (IP), porque la regla de introducción del generalizador la tenemos prohibida. Aquí es fundamental recordar que en la demostración de IP no se generaliza respecto a la variable que particularizamos. Notemos que es "de sentido común": si una variable procede de eliminar un $\bigvee u$, luego no podemos ligarla con un $\bigwedge u$, sino que tendremos que volver a introducir un particularizador.

Un uso incorrecto de EP Veamos ahora un ejemplo que explicite la necesidad de no generalizar respecto de variables libres en $\mathtt{S}^x_u\alpha$ tras haber aplicado EP. Usamos que $0 = 0$ es un teorema lógico, cosa que probaremos más abajo, cuando estudiemos las reglas de inferencia asociadas al igualador.

(1)	$0 = 0$	Teorema lógico
(2)	$\bigvee u\, u = 0$	IP 1
(3)	$x = 0$	EP 2
(4)	$\bigwedge u\, u = 0$	IG 3 (incorrecto)

Si la regla de eliminación del particularizador fuera válida sin la restricción de no usar IG respecto de la variable propia x, podríamos concluir que $\bigwedge u\, u = 0$ es un teorema lógico, pero no puede serlo porque es fácil construir un modelo en el que esta sentencia es falsa. ■

Ejemplo Como ilustración de las reglas que acabamos de presentar vamos a demostrar que existe una persona en el mundo tal que, si logra vivir hasta los 100 años, todos viviremos hasta los 100 años. Para ello consideramos un relator monádico C que se interprete como "vivirá hasta los 100 años", y queremos probar que $\bigvee u(Cu \to \bigwedge v\, Cv)$. En efecto:

(1)	$\bigwedge v\, Cv$	Hipótesis
(2)	Cx	Hipótesis
(3)	$Cx \to \bigwedge v\, Cv$	
(4)	$\bigvee u(Cu \to \bigwedge v\, Cv)$	IP 3
(5)	$\bigwedge v\, Cv \to \bigvee u(Cu \to \bigwedge v\, Cv)$	
(6)	$\neg\bigwedge v\, Cv$	Hipótesis
(7)	$\bigvee v\, \neg Cv$	NP 6
(8)	$\neg Cx$	EP 7
(9)	Cx	Hipótesis
(10)	$\bigwedge v\, Cv$	C 8, 9
(11)	$Cx \to \bigwedge v\, Cv$	
(12)	$\bigvee u(Cu \to \bigwedge v\, Cv)$	IP 11
(13)	$\neg\bigwedge v\, Cv \to \bigvee u(Cu \to \bigwedge v\, Cv)$	
(14)	$(\bigwedge v\, Cv \vee \neg\bigwedge v\, Cv) \to \bigvee u(Cu \to \bigwedge v\, Cv)$	Dil 5, 13
(15)	$\bigwedge v\, Cv \vee \neg\bigwedge v\, Cv$	TND
(16)	$\bigvee u(Cu \to \bigwedge v\, Cv)$	MP 14, 15

■

3.4.3 Reglas relacionadas con el igualador

Las reglas siguientes sólo tienen sentido para lenguajes con igualador.

Reglas de introducción y eliminación del igualador Estas reglas son la expresión en términos de reglas de inferencia de los axiomas de tipo 13 y se prueban sin mas que usar EP y MP:

(II) $\mathtt{S}^t_u\alpha \vdash \bigwedge u(u = t \to \alpha)$,

(EI) $\bigwedge u(u = t \to \alpha) \vdash \mathtt{S}^t_u\alpha$.

Regla de la identidad (I) $\vdash t = t$.

Demostración: Sea x una variable libre que no esté en t.

(1)	$x = t \to x = t$	Teorema lógico
(2)	$\bigwedge u(u = t \to u = t)$	IG 1
(3)	$\mathtt{S}^t_u(u = t)$	EI 2
(4)	$t = t$	R 3

Notemos que en la prueba se usa IG, pero respecto de una variable propia que podemos elegir, por lo que no hay problema en usar esta regla en contextos en los que no sea lícito generalizar respecto de ciertas variables.

Regla de la simetría de la identidad (SI): $t_1 = t_2 \vdash t_2 = t_1$.

Demostración:

(1)	$t_2 = t_2$	I
(2)	$\mathtt{S}_u^{t_2}(t_2 = u)$	R 1
(3)	$\bigwedge u(u = t_2 \to t_2 = u)$	II 2
(4)	$t_1 = t_2 \to t_2 = t_1$	EG 3
(5)	$t_1 = t_2$	Premisa
(6)	$t_2 = t_1$	MP 4, 5

Regla de la transitividad de la identidad (TI): $t_1 = t_2,\ t_2 = t_3 \vdash t_1 = t_3$.

Demostración:

(1)	$t_2 = t_3$	Premisa
(2)	$\bigwedge u(u = t_2 \to u = t_3)$	II 1
(3)	$t_1 = t_2 \to t_1 = t_3$	EG 2
(4)	$t_1 = t_2$	Premisa
(5)	$t_1 = t_3$	MP 3, 4

Regla de equivalencia entre términos idénticos (ETI):

$$t_1 = t_2,\ \mathtt{S}_u^{t_2}\alpha \vdash \mathtt{S}_u^{t_1}\alpha, \qquad t_1 = t_2 \vdash \mathtt{S}_u^{t_1}t = \mathtt{S}_u^{t_2}t.$$

Estas reglas dicen que si $t_1 = t_2$, todo lo que podamos decir de t_1 lo podemos decir de t_2 (y viceversa).

Demostración:

(1)	$\mathtt{S}_u^{t_2}\alpha$	Premisa
(2)	$\bigwedge u(u = t_2 \to \alpha)$	II 1
(3)	$t_1 = t_2 \to \mathtt{S}_u^{t_1}\alpha$	EG 2
(4)	$t_1 = t_2$	Premisa
(5)	$\mathtt{S}_u^{t_1}\alpha$	MP 3, 4

(1)	$t_1 = t_2$	Premisa
(2)	$\mathtt{S}_u^{t_2}t = \mathtt{S}_u^{t_2}t$	I
(2)	$\bigwedge u(u = t_2 \to \mathtt{S}_u^{t_1}t = t)$	II 2
(3)	$t_1 = t_2 \to \mathtt{S}_u^{t_1}t = \mathtt{S}_u^{t_2}t$	EG 3
(4)	$t_1 = t_2$	Premisa
(5)	$\mathtt{S}_u^{t_1}t = \mathtt{S}_u^{t_2}t$	MP 3, 4

3.4.4 Reglas relacionadas con el descriptor

Regla de las descripciones propias (DP): $\overset{1}{\bigvee} u\alpha \vdash \mathtt{S}_u^{u|\alpha}\alpha$.

Regla de las descripciones impropias (DI): $\neg\overset{1}{\bigvee} u\alpha \vdash u|\alpha = v|(v = v)$.

Se siguen de los axiomas de tipos 14 y 15 y del recíproco del teorema de deducción. Estas reglas expresan lo mismo que los axiomas correspondientes. La primera dice que si sabemos que existe un único u que cumple α, podemos afirmar que $u|\alpha$ cumple α, es decir, que $u|\alpha$ es precisamente el único u que cumple α, y entonces decimos que se trata de una *descripción propia.* La segunda regla dice que todas las descripciones impropias hacen referencia a un mismo objeto, al que siempre podemos referirnos con la descripción $v|(v = v)$.

3.5 Resultados adicionales

Variantes de los lenguajes de primer orden En la sección precedente hemos visto que es posible eliminar algunos conectores lógicos de la definición de lenguaje formal, así como el cuantificador existencial. Concretamente, si modificamos la definición del lenguaje formal para que incluya únicamente los signos lógicos $\neg, \to, \bigwedge$ y definimos

$$\alpha \vee \beta \equiv \neg\alpha \to \beta, \quad \alpha \wedge \beta \equiv \neg(\neg\alpha \vee \neg\beta), \quad \alpha \leftrightarrow \beta \equiv (\alpha \to \beta) \wedge (\beta \to \alpha),$$

así como $\bigvee u\alpha \equiv \neg\bigwedge\neg\alpha$, podemos reducir los axiomas lógicos a los de la primera tabla de la página siguiente. Esta versión es la más práctica en la mayoría de los usos de la lógica de primer orden, pues reduce los casos que hay que considerar en las definiciones y en los metateoremas sobre el cálculo deductivo de la lógica de primer orden (y en su semántica).

No obstante, para estudiar más a fondo la teoría de la demostración es preferible considerar lenguajes formales cuyos conectores primitivos sean $\neg, \vee$ y con los dos cuantificadores $\bigwedge, \bigvee$. Esto requiere definir

$$\alpha \to \beta \equiv \neg\alpha \vee \beta, \quad \alpha \wedge \beta \equiv \neg(\neg\alpha \vee \neg\beta), \quad \alpha \leftrightarrow \beta \equiv (\alpha \to \beta) \wedge (\beta \to \alpha)$$

y tomar como axiomas lógicos los de la segunda tabla de la página siguiente. Son unos pocos más, pero sucede que los lenguajes formales así definidos presentan un mayor grado de simetría que la teoría de la demostración sabe aprovechar muy fructíferamente.

Axiomas de $\boldsymbol{K_{\mathcal{L}}}$ con $\neg, \to, \bigwedge$

1. $\alpha \to (\beta \to \alpha)$,
2. $(\alpha \to (\beta \to \gamma)) \to ((\alpha \to \beta) \to (\alpha \to \gamma))$,
3. $(\neg\alpha \to \neg\beta) \to (\beta \to \alpha)$,
4. $\bigwedge u\alpha \to \mathtt{S}^t_u\alpha$,
5. $\bigwedge u(\alpha \to \beta) \to (\alpha \to \bigwedge u\beta)$.

   Sólo si $\mathcal{L}$ tiene igualador:

6. $\bigwedge u(u = t \to \alpha) \leftrightarrow \mathtt{S}^t_u\alpha$.

   Sólo si además $\mathcal{L}$ tiene descriptor:

7. $\overset{1}{\bigvee} u\alpha \to \mathtt{S}^{u|\alpha}_u\alpha$.
8. $\neg\overset{1}{\bigvee} u\alpha \to (u|\alpha) = v|(v = v)$.

Los axiomas 1, 2, 5 permiten probar el teorema de deducción y (junto con la regla MP) regulan el implicador, el axioma 3 regula el negador, el axioma 4 (junto con la regla IG) regula el generalizador, el axioma 6 regula el igualador y los axiomas 7 y 8 regulan el descriptor.

Axiomas de $K_{\mathcal{L}}$ con $\neg, \vee, \bigwedge, \bigvee$

1. $\alpha \to (\beta \to \alpha)$,
2. $(\alpha \to (\beta \to \gamma)) \to ((\alpha \to \beta) \to (\alpha \to \gamma))$,
3. $(\neg\alpha \to \neg\beta) \to (\beta \to \alpha)$,
4. $\alpha \to \alpha \vee \beta, \quad \beta \to \alpha \vee \beta$,
5. $(\alpha \to \gamma) \to ((\beta \to \gamma) \to (\alpha \vee \beta \to \gamma))$,
6. $\bigwedge u\alpha \to \mathtt{S}^t_u\alpha$,
7. $\bigwedge u(\alpha \to \beta) \to (\alpha \to \bigwedge u\beta)$.
8. $\bigvee u\, \alpha \leftrightarrow \neg\bigwedge u\neg\alpha$.

   Sólo si $\mathcal{L}$ tiene igualador:

9. $\bigwedge u(u = t \to \alpha) \leftrightarrow \mathtt{S}^t_u\alpha$.

   Sólo si además $\mathcal{L}$ tiene descriptor:

10. $\overset{1}{\bigvee} u\alpha \to \mathtt{S}^{u|\alpha}_u\alpha$.
11. $\neg\overset{1}{\bigvee} u\alpha \to (u|\alpha) = v|(v = v)$.

Hemos probado que los teoremas lógicos son los mismos en cualquier caso, es decir, que una fórmula es demostrable en una de las versiones del cálculo deductivo si y sólo si (reinterpretada adecuadamente) es demostrable en otra de ellas. La clave está en que a efectos prácticos nunca es relevante qué sucesión de signos concreta es una fórmula. Lo único que importa es que estén definidas razonablemente todas las fórmulas $\alpha \to \beta$, $\alpha \vee \beta$, etc., sin que importe si $\to$ es realmente un signo de $\mathcal{L}$ o una abreviatura.

Una cuestión más delicada es la posibilidad de eliminar la distinción entre variables libres y ligadas. En la práctica podemos prescindir de ella, en el sentido de que, por ejemplo, si pasamos de $\alpha(x)$ a $\bigvee u\, \alpha(u)$ por IG, no hay ningún problema en pasar de llamar x a una variable libre a llamar igualmente x a la variable ligada por la que la sustituimos al aplicar IG, y así pasamos de $\alpha(x)$ a $\bigwedge x\, \alpha(x)$, donde x es primero una variable libre y luego una variable ligada distinta. Puesto que en una deducción no pueden aparecer semifórmulas que no sean fórmulas, el contexto deja claro qué variables son libres y cuáles ligadas, sin necesidad de que usemos nombres distintos para unas y otras.

Ahora bien, también es posible eliminar incluso en teoría la distinción entre variables libres y ligadas, tal y como hacemos en los capítulos I y II de [LM], donde además hemos definido la sustitución de variables por términos sin exigir las condiciones de 3.11. Esto hace que el cálculo deductivo que en [LM] llamamos $K_{\mathcal{L}}$ no sea exactamente el mismo que hemos definido aquí, pero ambos son equivalentes.

En efecto, si $\mathcal{L}$ es un lenguaje formal de primer orden en el sentido definido aquí, llamaremos $\mathcal{L}_0$ al lenguaje formal que tiene las mismas variables, pero sin distinguir entre variables libres y ligadas, que es un lenguaje formal en el sentido definido en [LM]. Recíprocamente, dado un lenguaje $\mathcal{L}_0$ en el sentido de [LM], siempre podemos dividir sus variables en una familia de variables libres y otra de variables ligadas, con lo que obtenemos un lenguaje $\mathcal{L}$ en el sentido definido aquí, y se cumple lo siguiente:

Teorema 3.20 *Sea $\mathcal{L}$ un lenguaje formal de primer orden, y sea $\mathcal{L}_0$ el lenguaje formal en el sentido de* [LM] *que resulta de elimina la distinción entre las variables libres y ligadas de $\mathcal{L}$, con lo que toda fórmula de $\mathcal{L}$ lo es de $\mathcal{L}_0$. Entonces:*

1. *Toda fórmula de $\mathcal{L}_0$ cuyas variables libres sean variables libres de $\mathcal{L}$ es equivalente en $K_{\mathcal{L}_0}$ a una fórmula de $\mathcal{L}$.*
2. *Una fórmula γ de $\mathcal{L}$ es consecuencia en $K_{\mathcal{L}}$ de unas premisas $\gamma_0, \ldots, \gamma_r$ en $\mathcal{L}$ si y sólo si lo es en $K_{\mathcal{L}_0}$.*

Demostración: 1) es consecuencia del teorema [LM 2.16], que nos asegura que al sustituir las variables ligadas de la fórmula dada por otras variables nuevas obtenemos una fórmula equivalente, y si elegimos las nuevas variables entre las variables ligadas de $\mathcal{L}$ obtenemos una fórmula de $\mathcal{L}$.

Para probar 2) observamos, por una parte, que todo axioma de $K_{\mathcal{L}}$ lo es de $K_{\mathcal{L}_0}$, la regla de inferencia MP es la misma en ambos cálculos deductivos y, en cuanto a IG, tenemos que IG de $K_{\mathcal{L}}$ puede verse como una regla de inferencia derivada en $K_{\mathcal{L}_0}$, pues si la regla en $K_{\mathcal{L}}$ nos permite pasar de α a $\mathtt{S}^u_x\alpha$, en $K_{\mathcal{L}'}$ de α podemos pasar a $\bigwedge x\,\alpha$ (por IG), de aquí a $\mathtt{S}^u_x\alpha$ (por EG) y de aquí a $\bigwedge u\,\mathtt{S}^u_x\alpha$ (por IG). Esto implica que toda deducción en $K_{\mathcal{L}}$ puede completarse hasta una deducción en $K_{\mathcal{L}_0}$ con las mismas premisas y la misma conclusión, sin más que completar cada aplicación de IG como acabamos de indicar.

Supongamos ahora que $\alpha_0, \ldots, \alpha_s$ es una deducción en $K_{\mathcal{L}_0}$ a partir de las premisas $\gamma_0, \ldots, \gamma_r$. Pongamos que todas las variables que aparecen en la deducción están entre $x_0, \ldots, x_k$. Sean $y_0, \ldots, y_k$ variables libres de $\mathcal{L}$ distintas dos a dos, de modo que si x_i ya es una variable libre de $\mathcal{L}$, entonces $y_i \equiv x_i$. Por otro lado, sean $u_0, \ldots, u_k$ variables ligadas de $\mathcal{L}$ distintas dos a dos y distintas de las x_i.

Para cada fórmula α, llamamos α' a la fórmula que resulta de reemplazar cada ocurrencia libre de x_i por y_i y cada ocurrencia ligada de x_i por u_i. Vamos a ver que las fórmulas $\alpha'_0, \ldots, \alpha'_r$ forman una deducción en $K_{\mathcal{L}}$ con premisas $\gamma'_0, \ldots, \gamma'_s$. En efecto:

1. Si α_i es una premisa, entonces α'_i es una premisa.
2. Si α_i es un axioma de $K_{\mathcal{L}_0}$, se comprueba que α'_i es un axioma de $K_{\mathcal{L}}$.
3. Si α_i se deduce por MP de α_j y α_k, es inmediato que α'_i se deduce por MP de α'_j y α'_k.

4. Si $\alpha_i \equiv \bigwedge x\, \alpha_j$ se deduce por IG de α_j, entonces $\alpha'_i \equiv \bigwedge u\, \mathtt{S}^u_x \alpha'_j$, que se deduce por IG de α'_j en $K_{\mathcal{L}_0}$.

Ahora bien, como las premisas γ'_i y la conclusión γ son fórmulas de $\mathcal{L}$, todas las variables x_i que aparecen libres en ellas son variables libres de $\mathcal{L}$, luego no se alteran al calcular γ'_i o γ', es decir, que sólo modificamos las variables ligadas, por lo que estas transformadas son precisamente las fórmulas consideradas en el teorema [LM 2.16], y las equivalencias $\gamma_i \leftrightarrow \gamma'_i$ y $\gamma \leftrightarrow \gamma'$ son demostrables en $K_{\mathcal{L}}$ (porque la prueba del teorema [LM 2.16] vale también en $K_{\mathcal{L}}$ sin cambio alguno), y así podemos concluir que γ también es consecuencia de $\gamma_0, \ldots, \gamma_r$ en $K_{\mathcal{L}_0}$. ∎

Hemos probado este teorema para justificar que el cálculo deductivo que estamos considerando aquí es esencialmente el mismo considerado en [LM], pero, como ya hemos señalado, nosotros mantendremos siempre la distinción entre variables libres y ligadas porque esto es necesario para aplicar las técnicas del cálculo secuencial expuestas en [CS]. No obstante, en la práctica a partir de ahora usaremos letras arbitrarias para referirnos a variables libres y ligadas, sin excluir que demos el mismo nombre a una variable libre y a otra ligada (pero nunca a dos variables libres o a dos ligadas, salvo que lo advirtamos expresamente). ∎

Existencia con unicidad Generalizamos así la definición de la página 143:

$$\overset{1}{\bigvee} u_1 \cdots u_n \alpha \equiv \bigvee v_1 \cdots v_n \bigwedge u_1 \cdots u_n (\alpha \leftrightarrow u_1 = v_1 \wedge \cdots \wedge u_n = v_n),$$

donde $v_1, \ldots, v_n$ son variables (ligadas) que no aparezcan en α y sean distintas de $u_1, \ldots, u_n$.

Notemos que la elección de las variables $v_1, \ldots, v_n$ es irrelevante, pues dos elecciones distintas dan lugar a fórmulas equivalentes, pues siempre podemos eliminar los particularizadores sustituyendo las variables $v_1, \ldots, v_n$ por variables libres $x_1, \ldots, x_n$ que no estén en α y luego introducir particularizadores con otro juego de variables $v'_1, \ldots, v'_n$.

En general, este argumento vale para probar que cualquier fórmula de tipo $\bigvee u\alpha$ o $\bigwedge u\alpha$ es equivalente a la que resulta de cambiar la variable u por otra variable ligada cualquiera que no esté en α. Basta aplicar EP o EG y a continuación IP o IG.

La existencia con unicidad tiene una caracterización más útil en la práctica:

Teorema 3.21 *Si las variables $v_1, \ldots, v_n$ no están en $\alpha(u_1, \ldots, u_n)$ y son distintas de $u_1, \ldots, u_n$, entonces*

$$\vdash \overset{1}{\bigvee} u_1 \cdots u_n \alpha(u_1 \cdots u_n) \leftrightarrow \bigvee u_1 \cdots u_n\, \alpha(u_1, \ldots, u_n) \wedge$$

$$\bigwedge u_1 \cdots u_n v_1 \cdots v_n (\alpha(u_1, \ldots u_n) \wedge \alpha(v_1, \ldots, v_n) \to u_1 = v_1 \wedge \cdots \wedge u_n = v_n).$$

Nota Explicitando las sustituciones, la última parte del teorema se escribe

$$\bigwedge u_1 \cdots u_n v_1 \cdots v_n (\alpha \wedge S^{v_1 \cdots v_n}_{u_1 \cdots u_n} \alpha \to u_1 = v_1 \wedge \cdots \wedge u_n = v_n).$$

DEMOSTRACIÓN: Por simplicidad probamos el caso de una única variable:

(1)	$\bigvee u\alpha \wedge \bigwedge uv(\alpha \wedge S^v_u\alpha \to u = v)$	Hipótesis
(2)	$\mathtt{S}^y_u\alpha$	EC, EP 1
(3)	$\mathtt{S}^x_u\alpha$	Hipótesis
(4)	$\mathtt{S}^x_u\alpha \wedge \mathtt{S}^y_u\alpha \to x = y$	EC, EG, 1
(5)	$x = y$	IC 3, 2; MP 4
(6)	$\mathtt{S}^x_u\alpha \to x = y$	
(7)	$x = y$	Hipótesis
(8)	$\mathtt{S}^x_u\alpha$	EC EP 1
(9)	$x = y \to \mathtt{S}^x_u\alpha$	
(10)	$\mathtt{S}^x_u\alpha \leftrightarrow x = y$	IB 6, 9
(11)	$\bigwedge u(\alpha \leftrightarrow u = y)$	IG 10
(12)	$\bigvee v \bigwedge u(\alpha \leftrightarrow u = v)$	IP 11
(13)	$\overset{1}{\bigvee} u\alpha$	R 12
(14)	$\bigvee u\alpha \wedge \bigwedge uv(\alpha \wedge S^v_u\alpha \to u = v) \to \overset{1}{\bigvee} u\alpha$	

Por abreviar hemos aplicado varias reglas de inferencia simultáneamente y —aunque habíamos dicho que no lo íbamos a hacer— hemos distinguido las variables libres de las ligadas en la prueba). La implicación opuesta es similar:

(1)	$\overset{1}{\bigvee} u\alpha$	Hipótesis
(2)	$\bigvee v \bigwedge u(\alpha \leftrightarrow u = v)$	R 1
(3)	$\bigwedge u(\alpha \leftrightarrow u = z)$	EP 2
(4)	$\mathtt{S}^z_u\alpha \leftrightarrow z = z$	EG 3
(5)	$\mathtt{S}^z_u\alpha$	EB, I, MP, 4
(6)	$\bigvee u\, \alpha$	IP 5
(7)	$\mathtt{S}^x_u\alpha \wedge \mathtt{S}^y_u\alpha$	Hipótesis
(8)	$\mathtt{S}^x_u\alpha \to x = z$	EG, EB 3
(9)	$\mathtt{S}^y_u\alpha \to y = z$	EG, EB 3
(10)	$x = y$	EC 7, MP 8, MP 9, SI, TI
(11)	$\mathtt{S}^x_u\alpha \wedge \mathtt{S}^y_u\alpha \to u = v$	
(12)	$\bigwedge uv(\alpha \wedge \mathtt{S}^v_u\alpha \to u = v)$	IG 11
(13)	$\overset{1}{\bigvee} u\alpha \to \bigvee u\alpha \wedge \bigwedge uv(\alpha \wedge S^v_u\alpha \to u = v)$	

■

Disyunciones y conjunciones finitas Si $\mathcal{L}$ es un lenguaje formal con igualador (aunque esto no es esencial) y llamamos $0 \equiv \bigvee x\ x \neq x$, se cumple claramente:

1. $\bigwedge \alpha\beta \in \mathrm{Form}(\mathcal{L})\ \alpha \vee \beta \in A$
2. $\bigwedge \alpha\beta \in \mathrm{Form}(\mathcal{L})\ \vdash \alpha \vee \beta \leftrightarrow \beta \vee \alpha$

3. $\bigwedge\alpha\beta\gamma \in \mathrm{Form}(\mathcal{L})\ \vdash \alpha \vee (\beta \vee \gamma) \leftrightarrow (\alpha \vee \beta) \vee \gamma$
4. $\bigwedge\alpha \in \mathrm{Form}(\mathcal{L})\ \vdash \alpha \vee 0 \leftrightarrow \alpha$.

Estas propiedades son análogas a los presupuestos del último apartado de la sección 2.2 salvo que tenemos equivalencias lógicas en lugar de igualdades. Si $s : I_m \longrightarrow \mathrm{Form}(\mathcal{L})$, podemos considerar igualmente el funtor dado por

$$\bigvee_{i<0} s_i = 0, \qquad \bigwedge n < m \bigvee_{i<n+1} s_i = \bigvee_{i<n} s_i \vee s_n,$$

y los teoremas de dicho apartado se adaptan trivialmente cambiando igualdades por equivalencias lógicas. Por ejemplo, la tercera propiedad enunciada allí se convierte en la equivalencia

$$\vdash \bigvee_{i<m+n} s_i = \bigvee_{i<m} s_i \vee \bigvee_{i<n} s_{m+i}.$$

La cuarta propiedad (junto con su prueba) se traduce en este contexto a

$$f : I_n \longrightarrow I_n \to \vdash \bigvee_{i<n} s_i \leftrightarrow \bigvee_{i<n} s_{f(i)}.$$

A su vez, esta propiedad permite definir la disyunción

$$\bigvee_{\alpha\in x} \alpha$$

para cualquier conjunto finito de fórmulas x, de modo que

$$\bigvee_{\alpha\in\varnothing} \alpha = 0, \qquad \vdash \bigvee_{\alpha\in\{\beta\}} \alpha \leftrightarrow \beta,$$

así como que si x e y son conjuntos finitos disjuntos de fórmulas de $\mathcal{L}$, se cumple

$$\vdash \bigvee_{\alpha\in x\cup y} \alpha \leftrightarrow \bigvee_{\alpha\in x} \alpha \vee \bigvee_{\alpha\in y} \alpha.$$

Estos resultados justifican fácilmente el uso de notaciones más generales, como

$$\bigvee_{i=1}^{n} \alpha_i \qquad \text{o} \qquad \alpha_1 \vee \cdots \vee \alpha_n,$$

así como las manipulaciones elementales de estas expresiones (siempre en términos de equivalencias lógicas y no de igualdades).

De forma completamente análoga se pueden definir las conjunciones finitas, que podemos expresar con las notaciones

$$\bigwedge_{i=1}^{n} \alpha_i \qquad \text{o} \qquad \alpha_1 \wedge \cdots \wedge \alpha_n.$$

Notemos que la conjunción vacía $\bigwedge_{\alpha\in\varnothing} \alpha$ tiene que definirse como $1 = \bigvee x\ x = x$ para que se cumpla que $\vdash \alpha \wedge 1 \leftrightarrow \alpha$.

Algunas equivalencias El teorema siguiente permite probar que si en una fórmula sustituimos una subsemifórmula por otra equivalente obtenemos una fórmula equivalente:

Teorema 3.22 *Las equivalencias siguientes son teoremas lógicos:*

1. $(\alpha \leftrightarrow \alpha') \leftrightarrow (\neg\alpha \leftrightarrow \neg\alpha')$,
2. $\big((\alpha \leftrightarrow \alpha') \wedge (\beta \leftrightarrow \beta')\big) \rightarrow \big((\alpha \rightarrow \beta) \leftrightarrow (\alpha' \rightarrow \beta')\big)$,
3. $\big((\alpha \leftrightarrow \alpha') \wedge (\beta \leftrightarrow \beta')\big) \rightarrow \big((\alpha \vee \beta) \leftrightarrow (\alpha' \vee \beta')\big)$,
4. $\big((\alpha \leftrightarrow \alpha') \wedge (\beta \leftrightarrow \beta')\big) \rightarrow \big((\alpha \wedge \beta) \leftrightarrow (\alpha' \wedge \beta')\big)$,
5. $\big((\alpha \leftrightarrow \alpha') \wedge (\beta \leftrightarrow \beta')\big) \rightarrow \big((\alpha \leftrightarrow \beta) \leftrightarrow (\alpha' \leftrightarrow \beta')\big)$,
6. $\bigwedge u(\alpha \leftrightarrow \beta) \rightarrow (\bigwedge u\ \alpha \leftrightarrow \bigwedge u\ \beta)$,
7. $\bigwedge u(\alpha \leftrightarrow \beta) \rightarrow (\bigvee u\ \alpha \leftrightarrow \bigvee u\ \beta)$,
8. $\bigwedge u(\alpha \leftrightarrow \beta) \rightarrow (\overset{1}{\bigvee} u\ \alpha \leftrightarrow \overset{1}{\bigvee} u\ \beta)$.

DEMOSTRACIÓN: Veamos, por ejemplo, 6):

(1)	$\bigwedge u(\alpha \leftrightarrow \beta)$	Hipótesis
(2)	$\alpha \leftrightarrow \beta$	EG 1
(3)	$\bigwedge u\alpha$	Hipótesis
(4)	α	EG 3
(5)	β	MP 2, 5 (omitiendo EB)
(6)	$\bigwedge u\beta$	IG 5
(7)	$\bigwedge u\alpha \rightarrow \bigwedge u\beta$	
(8)	$\bigwedge u\beta \rightarrow \bigwedge u\alpha$	Se prueba análogamente
(9)	$\bigwedge u\alpha \leftrightarrow \bigwedge u\beta$	IB 7, 8

■

A partir de aquí es fácil probar que si en una expresión sustituimos una variable ligada por otra que no aparezca en la subfórmula que empieza con el cuantificador o el descriptor que la liga, obtenemos otra expresión equivalente (si es una fórmula) o igual (si es un término).

Por ejemplo, si en $\bigwedge uv(u \mid v \rightarrow \bigvee w(v = u \cdot w))$ cambiamos la w por otra variable ligada w', obtenemos una fórmula equivalente. El primer paso es probar la equivalencia

$$\bigvee w(y = x \cdot w) \leftrightarrow \bigvee w'(y = x \cdot w').$$

Para ello suponemos un miembro, eliminamos el cuantificador sustituyendo la variable w o w' por una variable libre z, y volvemos a introducir el cuantificador con la otra variable. En segundo lugar vamos aplicando sistemáticamente el teorema anterior, que nos da:

$$(x \mid y \rightarrow \bigvee w(y = x \cdot w)) \leftrightarrow (x \mid y \rightarrow \bigvee w'(y = x \cdot w')),$$

$$\bigwedge v((x \mid v \to \bigvee w(v = x \cdot w)) \leftrightarrow (x \mid v \to \bigvee w'(v = x \cdot w'))),$$
$$\bigwedge v(x \mid v \to \bigvee w(v = x \cdot w)) \leftrightarrow \bigwedge v(x \mid v \to \bigvee w'(v = x \cdot w')),$$
$$\bigwedge u(\bigwedge v(u \mid v \to \bigvee w(v = u \cdot w)) \leftrightarrow \bigwedge v(u \mid v \to \bigvee w'(v = u \cdot w'))),$$
$$\bigwedge uv(u \mid v \to \bigvee w(v = u \cdot w)) \leftrightarrow \bigwedge uv(u \mid v \to \bigvee w'(v = u \cdot w')).$$

Es por esto que, en la práctica, todas las elecciones de variables ligadas resultan irrelevantes, salvo por el hecho de que conflictos entre ellas. Por ejemplo, en el ejemplo anterior no podríamos haber cambiado w por v, lo que técnicamente se refleja en que, de haberlo hecho así, al introducir el cuantificador $\bigwedge v$ nos encontraríamos con que x no es sustituible por v. ∎

Forma prenexa Los lenguajes formales permiten definir una noción de "complejidad" de una afirmación que resulta útil en contextos muy variados. Es frecuente que a los estudiantes les cueste asimilar la noción de límite de una función en un punto más de lo que les cuesta comprender otros conceptos del mismo nivel. Uno de los factores que influyen en ello es que empieza más o menos así: *"Para todo $\epsilon > 0$ existe un $\delta > 0$ tal que para todo $x \in \mathbb{R}$, ..."*. La dificultad no está en que haya tres cuantificadores, pues una definición que empiece con *"Para todo ϵ, para todo δ y para todo x se cumple ..."* resulta mucho más sencilla. La complejidad de la definición de límite se debe a que los tres cuantificadores se alternan: *"para todo... existe... para todo ..."*

Vamos a definir la complejidad de una fórmula en términos de la alternancia de sus cuantificadores. Para ello introducimos la noción de forma prenexa:

Definición 3.23 Se dice que una fórmula sin descriptores α de un lenguaje formal $\mathcal{L}$ está en *forma prenexa* si $\alpha \equiv \pi\alpha_0$, donde α_0 es una semifórmula sin cuantificadores y π es una sucesión finita de cuantificadores universales $\bigwedge u$ y/o existenciales $\bigvee u$. A π se le llama *prefijo* de α. En tal caso, se dice que α es de tipo Σ_n (Π_n) si su prefijo consta de n bloques de cuantificadores alternados empezando por un cuantificador existencial (universal). Las fórmulas sin cuantificadores[11] se llaman fórmulas Δ_0.

Por ejemplo, una fórmula de tipo Σ_3 es

$$\bigvee xy \bigwedge uvw \bigvee z(x + u = y \wedge yvw = z).$$

Esta clasificación tiene interés porque, como veremos a continuación, toda fórmula es lógicamente equivalente a una fórmula en forma prenexa. La prueba se basa en el teorema siguiente, que dejamos como ejercicio:

[11] En realidad, el concepto de fórmula Δ_0 depende del contexto, aunque se trata siempre de una restricción sobre los cuantificadores que pueden aparecer en la fórmula. Aquí hemos considerado el caso extremo de no admitir cuantificadores, pero, por ejemplo, en aritmética se admite que contengan cuantificadores acotados, de la forma $\bigwedge u \leq x$, $\bigvee u \leq x$, mientras que en teoría de conjuntos se admiten cuantificadores $\bigwedge u \in x$, $\bigvee u \in x$ en las fórmulas Δ_0.

Teorema 3.24 *Se cumple:*

1. $\vdash (\alpha \to \bigwedge u\beta) \leftrightarrow \bigwedge u(\alpha \to \beta)$,
2. $\vdash (\alpha \to \bigvee u\beta) \leftrightarrow \bigvee u(\alpha \to \beta)$,
3. $\vdash (\bigwedge u\alpha \to \beta) \leftrightarrow \bigvee u(\alpha \to \beta)$,
4. $\vdash (\bigvee u\alpha \to \beta) \leftrightarrow \bigwedge u(\alpha \to \beta)$,
5. $\vdash (\alpha \vee \bigwedge u\beta) \leftrightarrow \bigwedge u(\alpha \vee \beta)$,
6. $\vdash (\alpha \vee \bigvee u\beta) \leftrightarrow \bigvee u(\alpha \vee \beta)$,
7. $\vdash (\bigwedge u\alpha \wedge \beta) \leftrightarrow \bigwedge u(\alpha \wedge \beta)$,
8. $\vdash (\bigvee u\alpha \wedge \beta) \leftrightarrow \bigvee u(\alpha \wedge \beta)$.

(Notemos que la variable ligada u no puede estar libre en las fórmulas que aparecen sin cuantificar.)

Ahora es fácil probar:

Teorema 3.25 *Para toda fórmula α, existe otra fórmula β en forma prenexa con las mismas variables libres que α tal que $\vdash \alpha \leftrightarrow \beta$.*

Demostración: Lo probamos por inducción[12] sobre la longitud de α.

Si $\alpha \equiv R_i^n t_1 \cdots t_n$, entonces ya está en forma prenexa, pues no tiene cuantificadores. Tomamos $\beta \equiv \alpha$.

Si $\alpha \equiv \neg\gamma$, por hipótesis de inducción sabemos que $\vdash \gamma \leftrightarrow \pi\delta$, para cierta fórmula $\pi\delta$ en forma prenexa, luego por el teorema 3.22 tenemos $\vdash \neg\gamma \leftrightarrow \neg\pi\delta$. Aplicando (NG) y (NP) podemos "meter" el negador, y así $\vdash \neg\gamma \leftrightarrow \pi'\neg\delta$, donde π' es la sucesión de cuantificadores que resulta de cambiar cada cuantificador universal de π por uno existencial y viceversa.

Si $\alpha \equiv \gamma \to \delta$, por hipótesis de inducción $\vdash \gamma \leftrightarrow \pi\epsilon$ y $\vdash \delta \leftrightarrow \pi'\eta$. Eliminando e introduciendo cuantificadores si es preciso, podemos suponer que las variables que liga π no están en $\pi'\eta$ y viceversa. Por 3.22 tenemos que $\vdash \alpha \leftrightarrow (\pi\epsilon \to \pi'\eta)$. Por el teorema anterior, $\vdash \alpha \leftrightarrow \pi'(\pi\epsilon \to \eta)$, y también $\vdash (\pi\epsilon \to \eta) \leftrightarrow \pi''(\epsilon \to \eta)$. Por (IG) y 3.22 tenemos que $\vdash \pi'(\pi\epsilon \to \eta) \leftrightarrow \pi'\pi''(\epsilon \to \eta)$ y, por lo tanto, $\vdash \alpha \leftrightarrow \pi'\pi''(\epsilon \to \eta)$.

Si $\alpha \equiv \bigwedge u\gamma(u)$, por hipótesis de inducción $\vdash \gamma(x) \leftrightarrow \pi\delta(x)$. Por (IG) tenemos que $\vdash \bigwedge u(\gamma(u) \leftrightarrow \pi\delta(u))$ y por 3.22 queda $\vdash \alpha \leftrightarrow \bigwedge u\pi\delta$.

Si α es de cualquier otra forma, se puede reducir a los casos anteriores. Por ejemplo, si $\alpha = \beta \vee \gamma$, sabemos que es equivalente a $\neg\beta \to \gamma$, y basta aplicar los casos del negador y del implicador, etc.

Es fácil comprobar que en cada caso las variables libres de la fórmula construida son las mismas que las de la fórmula dada. ∎

[12] Podemos definir un funtor que a cada semifórmula β le asigne una fórmula β^* que resulta de sustituir las variables ligadas de α que estén libres por unas variables libres nuevas que no estén en α. Fijada una fórmula α, en ARP podemos definir el conjunto de todas sus subsemifórmulas y, por reemplazo, el conjunto $S(\alpha)$ de todas las fórmulas β^* tales que β es una subsemifórmula de α A su vez, podemos definir $S_i(\alpha)$ como el conjunto de los elementos de $S(\alpha)$ de longitud i. La prueba del teorema consiste en definir un funtor diádico F de modo que $F(\alpha, i)$ sea una función de dominio $S_i(\alpha)$ que a cada α_0 le asigne un par $F(\alpha, i)(\alpha_0) = \langle\beta, d\rangle$ de modo que β sea una fórmula en forma prenexa con las mismas variables libres que α_0 y $\overset{d}{\vdash}(\alpha_0 \leftrightarrow \beta)$. Suponemos el funtor definido para $j < i$ y lo definimos para i.

En la práctica es fácil extraer los cuantificadores de cualquier fórmula dada usando el teorema 3.24.

3.6 Teorías axiomáticas

La definición de sistema deductivo formal permite considerar axiomas y reglas de inferencia arbitrarios. Sin embargo, demostraremos que $K_{\mathcal{L}}$ captura plenamente el razonamiento lógico tal y como los matemáticos lo entienden, luego lo único que necesitamos es añadirle axiomas que expresen propiedades específicas de objetos que queramos estudiar, sean números, conjuntos, figuras geométricas, etc. Ello nos lleva a la definición siguiente:

Definición 3.26 Una *teoría axiomática* (de primer orden) sobre un lenguaje formal $\mathcal{L}$ es un sistema deductivo formal T sobre $\mathcal{L}$ cuyos axiomas contengan a los de $K_{\mathcal{L}}$ y cuyas reglas de inferencia sean las de $K_{\mathcal{L}}$.

En toda teoría axiomática podemos distinguir entre sus *axiomas lógicos* (los axiomas de $K_{\mathcal{L}}$) y sus *axiomas propios* (los axiomas añadidos a los de $K_{\mathcal{L}}$). Normalmente, cuando se habla de los axiomas de una teoría axiomática, se sobreentiende que se trata de sus axiomas propios.

Así por ejemplo, la teoría de conjuntos ZFC, que sirve de fundamento a la mayor parte de la matemática actual, es una teoría axiomática en este sentido. Si $\mathcal{L}$ es el lenguaje formal de ZFC, lo que se conoce usualmente como "axiomas de ZFC" son los axiomas añadidos a los de $K_{\mathcal{L}}$ que especifican las propiedades que suponemos que cumplen los conjuntos abstractos, pero la lógica subyacente en ZFC no es ni más ni menos que la de $K_{\mathcal{L}}$.

Diremos que una teoría axiomática T' *extiende* a otra T si todos los signos del lenguaje de T son también signos del lenguaje de T' y todos los axiomas de T son teoremas de T'. Esto hace que, de hecho, todos los teoremas de T sean teoremas de T'. En particular, todas las teorías axiomáticas, en el sentido que acabamos de darle a este concepto, son extensiones de $K_{\mathcal{L}}$.

Observemos que todos los resultados que hemos demostrado para $K_{\mathcal{L}}$ son aplicables a cualquier teoría axiomática T, puesto que una demostración en T es lo mismo que una deducción en $K_{\mathcal{L}}$ que tiene por premisas los axiomas propios de T que aparezcan en ella.

Una teoría axiomática T es *contradictoria* si existe una fórmula α tal que $\underset{T}{\vdash} \alpha$ y $\underset{T}{\vdash} \neg\alpha$. En caso contrario se dice que T es *consistente.*

En virtud de la regla de contradicción (C), si T es una teoría axiomática contradictoria sobre un lenguaje formal $\mathcal{L}$, todas las fórmulas de $\mathcal{L}$ son teoremas de T, lo que significa que demostrar algo en T no aporta nada. Equivalentemente, una teoría es consistente si y sólo si existe al menos una fórmula que no es demostrable en T.

Es obvio que si una teoría axiomática T' extiende a otra T y T es contradictoria, también lo es T'. En particular, si $K_{\mathcal{L}}$ fuera contradictorio también lo serían todas las teorías axiomáticas. Afortunadamente, no es el caso:

Teorema 3.27 *Si $\mathcal{L}$ es cualquier lenguaje formal, $K_{\mathcal{L}}$ es consistente.*

DEMOSTRACIÓN: A lo largo de todo este capítulo hemos distinguido los resultados sobre modelos porque no son formalizables en ARP, pero esto no es exactamente así. Es fácil formalizar en ARP todo cuanto hemos dicho sobre modelos, satisfacción, verdad, etc. siempre y cuando exijamos que los modelos considerados tengan universo finito. No hemos insistido en ello porque no es algo muy útil, ya que prácticamente todos los modelos de interés tienen universo infinito, pero, dado cualquier lenguaje formal $\mathcal{L}$, siempre podemos definir (en ARP) un modelo de $\mathcal{L}$ cuyo universo sea $M_0 = \{0\}$, en el que todas las constantes de $\mathcal{L}$ se interpretan como 0, en el que todos los relatores se interpretan como las relaciones que se cumplen siempre, y en las que todos los funtores se interpretan como las funciones que toman el único valor posible, 0.

El teorema de corrección implica entonces que si $\vdash \alpha$, entonces $M_0 \vDash \alpha$, pero no puede ocurrir que $M_0 \vDash \alpha$ y $M_0 \vDash \neg\alpha$, por lo que no puede ocurrir que $K_{\mathcal{L}}$ sea contradictorio. ∎

Nota Aunque ARP no es una teoría axiomática en el sentido que hemos definido aquí, tiene sentido plantearse igualmente si es consistente o contradictoria, y el teorema de corrección para ARP prueba que, de hecho, es consistente, pues todo teorema de ARP tiene que ser verdadero (en el único modelo de ARP que hemos considerado) y una fórmula y su negación no pueden ser ambas verdaderas. Sin embargo, ARP no admite modelos finitos, y no es posible demostrar en ARP la consistencia de ARP, cualquier prueba empleará necesariamente técnicas no estrictamente finitistas. ∎

Definición 3.28 Diremos que una teoría axiomática T' es una *extensión conservativa* de otra teoría T si T' extiende a T y una fórmula del lenguaje de T es demostrable en T si y sólo si es demostrable en T'. Si además toda fórmula del lenguaje de T' es equivalente en T' a una del lenguaje de T (con a lo sumo las mismas variables libres), diremos que la extensión es *intrascendente.*

Observemos que si T' es una extensión conservativa de T y T es consistente, entonces T' también lo es.

Si T' es una extensión intrascendente de T, tenemos que en el lenguaje de T' no puede expresarse nada que no pueda expresarse ya en el lenguaje de T, por lo que tampoco puede demostrarse nada que no pueda demostrarse ya en T. Así pues, la capacidad expresiva y probativa de T' es la misma que la de T.

Esta situación se da cuando partimos de una teoría axiomática T sobre un lenguaje formal $\mathcal{L}$ con igualador, pero sin descriptor, y pasamos a considerar la teoría T' sobre el lenguaje $\mathcal{L}'$ que resulta de añadirle a $\mathcal{L}$ un descriptor (y, por consiguiente, a los axiomas de T, añadimos, no sólo los dos grupos de axiomas asociados al descriptor, sino también todos los axiomas de los demás tipos correspondientes a fórmulas de $\mathcal{L}'$ con descriptores). En realidad veremos que hace falta añadir un axioma más.

Eliminación de descriptores El descriptor complica sustancialmente la gramática de los lenguajes formales, pero vamos a probar que, aunque en la práctica es muy útil para formalizar definiciones, en teoría podemos prescindir de él bajo hipótesis muy débiles:

Teorema 3.29 (Eliminación de descriptores) *Sea T una teoría axiomática sobre un lenguaje formal $\mathcal{L}$ con igualador, pero sin descriptor, y llamemos $\mathcal{L}'$ al lenguaje formal que resulta de añadirle a $\mathcal{L}$ un descriptor. Supongamos que existe una fórmula $\phi(x)$ de $\mathcal{L}$, con x como única variable libre, tal que*

$$\underset{T}{\vdash} \overset{1}{\bigvee} u\, \phi(u).$$

Llamamos T' a la teoría axiomática sobre $\mathcal{L}'$ cuyos axiomas propios son los de T más el axioma $\phi(v|v = v)$. Entonces T' es una extensión intrascendente de T.

Observemos que si el lenguaje $\mathcal{L}$ tiene una constante c, siempre podemos tomar $\phi(x) \equiv x = c$. Otro caso se da en cualquier teoría de conjuntos, donde podemos tomar $\phi(x) \equiv \bigwedge u\, u \notin x$.

Así, el axioma que le hemos añadido a T' expresa que toda descripción impropia (todo lo que no esté bien definido) es, por definición, la constante c (o el conjunto vacío en una teoría de conjuntos).

Observemos que, por la unicidad,

$$\underset{T'}{\vdash} x = (v|v = v) \leftrightarrow \phi(x).$$

Teorema 3.30 *Si x no está en $u|\alpha$, se cumple*

$$\vdash x = u|\alpha \leftrightarrow \bigwedge u(\alpha \leftrightarrow u = x) \vee (\neg \overset{1}{\bigvee} u\, \alpha \wedge x = v|v = v).$$

Demostración: Esbozamos la prueba. Bajo la hipótesis $x = u|\alpha$, distinguimos dos casos, o bien $\overset{1}{\bigvee} u\ \alpha$ o bien $\neg \overset{1}{\bigvee} u\ \alpha$.

En el primer caso, por la definición de unicidad, tenemos $\bigvee v \bigwedge u(\alpha \leftrightarrow u = v)$. Eliminamos el particularizador, con lo que $\bigwedge u(\alpha \leftrightarrow u = y)$. Eliminando el generalizador llegamos a $S_u^{u|\alpha}\alpha \leftrightarrow (u|\alpha) = y$, pero la parte izquierda la tenemos por la regla de las descripciones propias, con lo que $y = u|\alpha$. Por hipótesis, $x = y$ y por la equivalencia de términos idénticos $\bigwedge u(\alpha \leftrightarrow u = x)$.

En el segundo caso, la regla de las descripciones impropias nos permite afirmar que $u|\alpha = v|v = v$ y por hipótesis $x = u|\alpha$, lo que nos lleva a la conclusión.

Supongamos ahora la parte derecha del teorema. Por la regla del dilema basta probar que ambas disyuntivas nos llevan a $x = u|\alpha$.

Si suponemos $(*)$: $\bigwedge u(\alpha \leftrightarrow u = x)$, introduciendo un particularizador obtenemos $\overset{1}{\bigvee} u\, \alpha$, luego la regla de las descripciones propias nos da $\mathtt{S}_u^{u|\alpha}\alpha$. Por otro lado, eliminando el generalizador en $(*)$ obtenemos $\mathtt{S}_u^{u|\alpha}\alpha \leftrightarrow u|\alpha = x$, luego concluimos que $x = u|\alpha$.

Si suponemos $\neg \overset{1}{\bigvee} u\, \alpha \wedge x = v|v = v$, la regla de las descripciones impropias nos da que $u|\alpha = v|v = v$, luego concluimos igualmente que $x = u|\alpha$. ∎

Combinando esto con la observación previa al teorema, vemos que

$$\underset{T'}{\vdash} x = u|\alpha \leftrightarrow \bigwedge u(\alpha \leftrightarrow u = x) \vee (\neg \overset{1}{\bigvee} u\, \alpha \wedge \phi(x)).$$

A continuación definimos, para cada semitérmino t de $\mathcal{L}'$, una semifórmula $\psi_t(x)$ de $\mathcal{L}$ que tenga las mismas variables libres que t más una variable libre adicional x y, para cada semifórmula α de $\mathcal{L}'$, una semifórmula α^* de $\mathcal{L}$ que tenga sus mismas variables libres (de modo que, si partimos de una expresión, obtenemos una fórmula).

La definición es la siguiente:

1. Si $t \equiv y$ es una variable (libre o ligada), entonces $\psi_t(x) \equiv x = y$.
2. Si $t \equiv c$ es una constante, $\psi_t(x) \equiv x = c$.
3. Si $t \equiv f^n(t_1, \ldots, t_n)$, entonces
$$\psi_t(x) \equiv \bigvee u_1 \cdots u_n(\psi_{t_1}(u_1) \wedge \cdots \wedge \psi_{t_n}(u_n) \wedge x = f^n(u_1, \ldots, u_n)).$$
4. Si $\alpha \equiv R^n(t_1, \ldots, t_n)$, entonces
$$\alpha^* \equiv \bigvee u_1 \cdots u_n(\psi_{t_1}(u_1) \wedge \cdots \wedge \psi_{t_n}(u_n) \wedge R^n(u_1, \ldots, u_n)).$$
5. Si $\alpha \equiv \neg\beta$, entonces $\alpha^* \equiv \neg\beta^*$.
6. Si $\alpha \equiv \beta \vee \gamma$, entonces $\alpha^* \equiv \beta^* \vee \gamma^*$, e igualmente para los demás conectores.
7. Si $\alpha \equiv \bigwedge u\, \beta$, entonces $\alpha^* \equiv \bigwedge u\, \beta^*$.
8. Si $\alpha \equiv \bigvee u\, \beta$, entonces $\alpha^* \equiv \bigvee u\, \beta^*$.
9. Si $t \equiv u|\alpha$, entonces
$$\psi_t(x) \equiv \bigwedge u(\alpha^* \leftrightarrow u = x) \vee (\neg \overset{1}{\bigvee} u\, \alpha^* \wedge \phi(x)).$$

Necesitamos probar varios hechos sobre estas fórmulas.

1. Si x es una variable que no está en el término t de $\mathcal{L}'$ y α es una fórmula de $\mathcal{L}'$,
$$\underset{T'}{\vdash} \psi_t(x) \leftrightarrow x = t, \qquad \underset{T'}{\vdash} \alpha^* \leftrightarrow \alpha.$$
Y si t o α no tienen descriptores, la equivalencia correspondiente puede probarse en T.

   En efecto, razonando por inducción sobre la longitud de una expresión,[13] todos los casos son inmediatos, salvo el correspondiente a las descripciones $t \equiv u|\alpha$, donde usamos
$$\underset{T'}{\vdash} x = u|\alpha \leftrightarrow \bigwedge u(\alpha \leftrightarrow u = x) \vee (\neg \overset{1}{\bigvee} u\, \alpha \wedge \phi(x)), \quad \underset{T'}{\vdash} \alpha^*(x) \leftrightarrow \alpha(x).$$

[13] Técnicamente el planteamiento es el mismo que se indica en la nota al pie de la demostración del teorema 3.25.

2. $\underset{T}{\vdash} \overset{1}{\bigvee} v\, \psi_t(v)$.

   Nuevamente, todos los casos son inmediatos salvo el correspondiente a $t \equiv u|\alpha$. Distinguimos dos casos: o bien $\overset{1}{\bigvee} u\, \alpha^*(u)$, o bien $\neg \overset{1}{\bigvee} u\, \alpha^*(u)$. En el primer caso $\psi_t(x)$ sólo lo cumple el único x que cumple $\alpha^*(x)$ y en el segundo caso sólo lo cumple el único x que cumple $\phi(x)$.

3. Si t, t' son términos de $\mathcal{L}$, α es una fórmula y x es una variable que no esté en $\mathtt{S}^t_y t'$ ni en $\mathtt{S}^t_y \alpha$, entonces

   $$\underset{T}{\vdash} \psi_{\mathtt{S}^t_y t'}(x) \leftrightarrow \bigvee u(\psi_t(u) \wedge \mathtt{S}^u_y \psi_{t'}(x)), \quad \underset{T}{\vdash} (S^t_y \alpha)^* \leftrightarrow \bigvee u(\psi_t(u) \wedge \mathtt{S}^u_y \alpha^*).$$

   Probamos por inducción sobre la longitud de una expresión θ que cumple lo requerido para t' o α según si es un término o una fórmula.

   Si $\theta \equiv z$ es una variable, distinguimos dos casos, según si $y \equiv z$ o $y \not\equiv z$. En el primer caso hay que probar (en T) que

   $$\psi_t(x) \leftrightarrow \bigvee u(\psi_t(u) \wedge x = u),$$

   lo cual es obvio.

   En el segundo caso hay que probar

   $$x = z \leftrightarrow \bigvee u(\psi_t(u) \wedge x = z),$$

   y esto es también un teorema porque, según el apartado precedente, en T podemos probar que $\bigvee u \psi_t(u)$.

   El caso en que $\theta \equiv c$ es una constante es idéntico a la segunda parte del caso anterior, cambiando z por c.

   Si $\theta \equiv f^n(t_1, \ldots, t_n)$, llamando $t'_i \equiv \mathtt{S}^t_y t_i$, tenemos que

   $$\psi_{\mathtt{S}^t_y \theta}(x) \equiv \bigvee u_1 \cdots u_n(\psi_{t'_1}(u_1) \wedge \cdots \wedge \psi_{t'_n}(u_n) \wedge x = f^n(u_1, \ldots, u_n)).$$

   Por hipótesis de inducción,

   $$\psi_{t'_i}(x_i) \leftrightarrow \bigvee v_i(\psi_t(v_i) \wedge \mathtt{S}^{v_i}_y \psi_{t_i}(x_i)),$$

   pero en T se prueba $\overset{1}{\bigvee} v\, \psi_t(v)$, luego todos los v_i tienen que ser iguales, y $\psi_{t'_i}(x_i)$ equivale a

   $$\bigvee u(\psi_t(u) \wedge \mathtt{S}^u_y \bigvee u_1 \cdots u_n(\psi_{t_1}(u_1) \wedge \cdots \wedge \psi_{t_n}(u_n) \wedge x = f^n(u_1, \ldots, u_n))),$$

   que es lo mismo que $\bigvee u(\psi_t(u) \wedge \mathtt{S}^u_y \psi_\theta(x))$.

   Si $\theta \equiv R^n(t_1, \ldots, t_n)$, llamando $t' \equiv \mathtt{S}^t_y t_i$, como antes, tenemos que

   $$\theta^* \equiv \bigvee u_1 \cdots u_n(\psi_{t'_1}(u_1) \wedge \cdots \wedge \psi_{t'_n}(u_n) \wedge R^n(u_1, \ldots, u_n))$$

y, aplicando como antes la hipótesis de inducción, esto equivale a

$$\bigvee u(\psi_t(u) \wedge \mathtt{S}_y^u \bigvee u_1 \cdots u_n(\psi_{t_1}(u_1) \wedge \cdots \wedge \psi_{t_n}(u_n) \wedge R^n(u_1, \ldots, u_n))),$$

que es lo mismo que $\bigvee u(\psi_t(u) \wedge \mathtt{S}_y^u \theta^*)$.

Si $\theta \equiv \neg\alpha$, por hipótesis de inducción en T se prueba

$$(S_y^t \alpha)^* \leftrightarrow \bigvee u(\psi_t(u) \wedge \mathtt{S}_y^u \alpha^*).$$

Teniendo en cuenta que también tenemos $\overset{1}{\bigvee} u \psi_t(u)$, es fácil concluir que

$$\neg(S_y^t \alpha)^* \leftrightarrow \bigvee u(\psi_t(u) \wedge \mathtt{S}_y^u \neg\alpha^*).$$

Los casos correspondientes a los demás conectores se tratan análogamente (siempre teniendo en cuenta la unicidad de ψ_t).

Si $\theta \equiv \bigwedge v\, \alpha$, en el caso en que $y \equiv v$, hay que probar que

$$\theta^* \leftrightarrow \bigvee u(\psi_t(u) \wedge \theta^*),$$

lo cual se cumple porque en T se prueba $\bigvee u\, \psi_t(u)$.

En caso contrario, aplicando la hipótesis de inducción (y cambiando la variable u por otra, si hace falta, para que $u \not\equiv v$):

$$(\mathtt{S}_y^t \theta)^* \equiv \bigwedge v(\mathtt{S}_y^t \alpha^*) \leftrightarrow \bigwedge v \bigvee u(\psi_t(u) \wedge \mathtt{S}_y^u \alpha^*),$$

y esto equivale a $\bigvee u(\psi_t(u) \wedge \mathtt{S}_y^u \bigwedge v \alpha^*)$, que es $\bigvee u(\psi_t(u) \wedge \mathtt{S}_y^u \theta^*)$.

El caso en que $\theta \equiv \bigvee v\alpha$ es análogo.

Si $\theta \equiv v|\alpha$, el caso en que $y \equiv v$ o y no está libre en α es idéntico a la parte correspondiente del caso anterior. En caso contrario,

$$\psi_{\mathtt{S}_y^t \theta}(x) \equiv \bigwedge v((\mathtt{S}_y^t \alpha)^* \leftrightarrow v = x) \vee (\neg \overset{1}{\bigvee} v(\mathtt{S}_y^t \alpha)^* \wedge \phi(x)).$$

Aplicando la hipótesis de inducción, esto equivale a

$$\bigwedge v(\bigvee u(\psi_t(u) \wedge \mathtt{S}_y^u \alpha^*) \leftrightarrow v = x) \vee (\neg \overset{1}{\bigvee} v \bigvee u(\psi_t(u) \wedge \mathtt{S}_y^u \alpha^*) \wedge \phi(x)).$$

Usando una vez más que en T se prueba que $\overset{1}{\bigvee} u\, \psi_t(u)$, esto equivale a

$$\bigvee u(\psi_t(u) \wedge (\bigwedge v(\mathtt{S}_y^u \alpha^* \leftrightarrow v = x) \vee (\neg \overset{1}{\bigvee} v\, \mathtt{S}_y^u \alpha^* \wedge \phi(x)))),$$

que a su vez equivale a $\bigvee u(\psi_t(u) \wedge \mathtt{S}_y^u \psi_\theta(x))$.

4. Si θ es un axioma de $K_{\mathcal{L}'}$, entonces $\underset{T}{\vdash} \theta^*$.

   Esto es inmediato para los axiomas del cálculo proposicional, pues al eliminar sus descriptores obtenemos otro axioma del mismo tipo. Por ejemplo,

   $$(\alpha \to (\beta \to \alpha))^* \equiv \alpha^* \to (\beta^* \to \alpha^*).$$

   Analizamos únicamente los casos en los que la comprobación no es inmediata:

   Si $\theta \equiv \bigwedge u\alpha \to \mathtt{S}^t_u\alpha$, entonces

   $$\theta^* \equiv \bigwedge u\alpha^* \to (S^t_u\alpha)^*),$$

   y por el punto precedente esto equivale en T a

   $$\bigwedge u\alpha^* \to \bigvee v(\psi_t(v) \wedge \mathtt{S}^v_u\alpha^*),$$

   y es fácil probar esto en T teniendo en cuenta que podemos probar $\bigvee v\, \psi_t(v)$.

   Si $\theta \equiv \bigwedge u(u = t \to \alpha) \leftrightarrow \mathtt{S}^t_u\alpha$, observemos en primer lugar que

   $$(x = t)^* \equiv \bigvee uv(u = x \wedge \psi_t(v) \wedge u = v),$$

   que equivale en T a $\psi_t(x)$. Por lo tanto, θ^* es equivalente a

   $$\bigwedge u(\psi_t(u) \to \alpha^*) \leftrightarrow \bigvee v(\psi_t(v) \wedge \mathtt{S}^v_u\alpha^*),$$

   y esto es ciertamente un teorema de T.

   Por ejemplo, si suponemos el miembro derecho, tomamos un x que cumpla $\psi_t(x) \wedge \alpha^*(x)$. Si se cumple $\psi_t(y)$, por la unicidad de ψ_t, tiene que ser $y = x$, luego también se cumple $\alpha^*(y)$, lo que nos da la implicación $\bigwedge u(\psi_t(u) \to \alpha^*)$. El recíproco es más sencillo.

   Si $\theta \equiv \overset{1}{\bigvee} u\, \alpha \to \mathtt{S}^{u|\alpha}_u\alpha$, entonces

   $$\theta^* \equiv \overset{1}{\bigvee} u\, \alpha^* \to \bigvee u(\psi_{u|\alpha}(u) \wedge \alpha^*),$$

   donde

   $$\psi_{u|\alpha}(x) \equiv \bigwedge u(\alpha^* \leftrightarrow u = x) \vee (\neg \overset{1}{\bigvee} u\, \alpha^* \wedge \phi(x)).$$

   Por lo tanto, si suponemos que existe un único u que cumple α^*, como también sabemos que existe un único x que cumple $\psi_{u|\alpha}(x)$, dicho x cumple la primera fórmula de la disyunción, de donde se sigue que cumple $\alpha^*(x)$, luego tenemos que $\bigvee u(\psi_{u|\alpha} \wedge \alpha^*)$.

   Por último, si $\theta \equiv \neg \overset{1}{\bigvee} u\, \alpha \to u|\alpha = v|v = v$, entonces θ^* equivale a

   $$\neg \overset{1}{\bigvee} u\, \alpha^* \to \bigvee u_1u_2(\psi_{u|\alpha}(u_1) \wedge \psi_{v|v=v}(u_2) \wedge u_1 = u_2),$$

que a su vez equivale a

$$\neg\overset{1}{\bigvee}u\,\alpha^* \to \bigvee u(\psi_{u|\alpha}(u) \wedge \psi_{v|v=v}(u)).$$

Ahora bien, bajo la hipótesis $\neg\overset{1}{\bigvee}u\,\alpha^*$, en T podemos razonar que el único x que cumple $\psi_{u|\alpha}(x)$ es también el único x que cumple $\phi(x)$. Por otro lado tenemos que distinguir dos casos, según si $\overset{1}{\bigvee}v\,v = v$ o no. En el primer caso todo es igual a todo, y el único x que cumple $\psi_{u|\alpha}(x)$ es también el único x que cumple $\psi_{v|v=v}(x)$, mientras que en el segundo caso se razona que el único x que cumple $\psi_{v|v=v}(x)$ es también el único x que cumple $\phi(x)$. En ambos casos tenemos que $\psi_{u|\alpha}(x) \wedge \psi_{v|v=v}(x)$, de donde obtenemos la conclusión.

5. Si θ es un axioma de T', entonces $\underset{T}{\vdash}\theta^*$.

   En efecto, ya lo hemos probado para los axiomas lógicos, también es cierto para los axiomas de T, pues son fórmulas de $\mathcal{L}$ y entonces θ^* equivale a θ en T, luego θ^* es un teorema. Sólo falta probarlo para el axioma $\theta \equiv \phi(v|v=v) \equiv \mathtt{S}_x^{v|v=v}\phi(x)$. Sabemos entonces que θ^* equivale a

   $$\bigvee u(\psi_{v|v=v}(u) \wedge \phi(u)),$$

   y en el apartado anterior ya hemos visto que esto es un teorema de T, pues, tanto si $\overset{1}{\bigvee}v\,v = v$ como si no, se cumple que el único x que cumple $\psi_{v|v=v}(x)$ coincide con el único x que cumple $\phi(x)$.

Ahora ya es inmediata la prueba del teorema 3.29: La primera parte la hemos demostrado ya, pues toda fórmula α de $\mathcal{L}'$ es equivalente en T' a la fórmula α^*. Para probar la segunda parte, suponemos que $\alpha_1, \ldots, \alpha_m$ es una demostración en T' de una fórmula $\alpha \equiv \alpha_m$ de $\mathcal{L}$. Basta probar inductivamente que cada α_i^* es un teorema de T, pues en particular tendremos que α^* es un teorema de T, y α^* es equivalente a α en T.

Si α_i es un axioma de T', ya hemos visto que α_i^* es un teorema de T. Si α_i se deduce de las líneas anteriores α_j y $\alpha_j \to \alpha_i$ por MP y α_j^* y $\alpha_j^* \to \alpha_i^*$ son teoremas de T, es obvio que α_i^* también lo es.

Por último, si $\alpha_i \equiv \bigwedge u\mathtt{S}_x^u\alpha_j$ se deduce por IG, entonces α_i^* es equivalente a $\bigwedge u\mathtt{S}_x^u\alpha_j^*$ y también se deduce por IG del teorema α_j^*. ∎

En particular, si una teoría axiomática T sobre un lenguaje sin descriptor es consistente, sigue siéndolo si le añadimos un descriptor (y un axioma de tipo $\phi(v|v=v)$ que determine la descripción impropia).

Capítulo IV

Teorías aritméticas

En el capítulo precedente hemos definido el concepto general de "teoría axiomática de primer orden", pero sucede que la aritmética recursiva primitiva que hemos usado como teoría básica para formalizar en ella la lógica, no es una teoría axiomática de primer orden: en efecto, el lenguaje de ARP carece de cuantificadores (y de conectores lógicos, aunque hemos visto que éstos pueden ser definidos aritméticamente, al igual que los cuantificadores acotados).

Aquí presentaremos una teoría axiomática de primer orden a la que llamaremos (provisionalmente) ARP^+ que, en cierto sentido que tendremos que precisar, es equivalente a la aritmética recursiva primitiva. Así, ARP^+ nos permitirá trabajar (más cómodamente) con la lógica de primer orden, que carece de las restricciones que impone la lógica de ARP, sin perder la identificación que proporciona ARP de los resultados que podemos demostrar con técnicas estrictamente finitistas. Después introduciremos otras teorías aritméticas (es decir, teorías diseñadas para hablar de los números naturales) estrechamente relacionadas con ARP y que nos permitirán relacionarla con la llamada Aritmética de Peano (AP).

Todo el contenido de este capítulo es formalizable —y, más aún, lo podemos considerar formalizado— en ARP. Por simplicidad consideraremos que los lenguajes formales tienen únicamente los conectores $\neg$ y $\vee$, de modo que los otros tres se definen a partir de ellos, pero mantendremos los dos cuantificadores como signos primitivos, a pesar de que podríamos eliminar el particularizador.

4.1 ARP como teoría de primer orden

En el capítulo precedente hemos definido el lenguaje que hemos llamado $\mathcal{L}^+_{\mathrm{arp}}$, que es el lenguaje con igualador —pero sin descriptor, pues sus funtores permiten hacer referencia a cualquier concepto que podamos definir sin necesidad del descriptor— cuyos únicos signos eventuales son la constante 0 y los funtores de $\mathcal{L}_{\mathrm{arp}}$ (que ahora consideramos como signos individuales en lugar de como cadenas de signos, pero esto es un tecnicismo irrelevante).

Si tomamos también como variables libres de $\mathcal{L}^+_{\text{arp}}$ las variables de $\mathcal{L}_{\text{arp}}$, entonces todos los signos de $\mathcal{L}_{\text{arp}}$ son signos de $\mathcal{L}^+_{\text{arp}}$, por lo que podemos considerar a todos los términos y fórmulas de $\mathcal{L}_{\text{arp}}$ como términos y fórmulas[1] de $\mathcal{L}^+_{\text{arp}}$. En particular, podemos ver todos los axiomas de ARP como fórmulas de $\mathcal{L}^+_{\text{arp}}$.

Definimos ahora una teoría axiomática sobre este lenguaje $\mathcal{L}^+_{\text{arp}}$. Recordemos que esto significa que sus únicas reglas de inferencia son las de $K_{\mathcal{L}^+_{\text{arp}}}$ y cuyos axiomas incluyen a los de $K_{\mathcal{L}^+_{\text{arp}}}$, por lo que sólo tenemos que especificar los axiomas propios:

Definición 4.1 Llamaremos *aritmética recursiva primitiva* (de primer orden) a la teoría axiomática ARP^+ cuyo lenguaje formal es $\mathcal{L}^+_{\text{arp}}$ y cuyos axiomas son los axiomas de ARP (es decir, las definiciones de los funtores) más los que indicamos a continuación:

1. $Sx \neq 0$,
2. $Sx = Sy \to x = y$,
3. $\alpha(0) \wedge \bigwedge u(\alpha(u) \to \alpha(Su)) \to \alpha(x)$,

para toda fórmula sin cuantificadores $\alpha(x)$ (tal vez con más variables libres).

Observemos que 3. no es un axioma, sino un esquema axiomático, es decir, un criterio que determina infinitas fórmulas con esta estructura, una para cada fórmula α. A dicho esquema se le conoce como *principio de inducción abierta.*[2]

El teorema siguiente nos da la conexión más obvia entre ARP y ARP^+:

Teorema 4.2 *Todo teorema de* ARP *es también un teorema de* ARP^+.

Demostración: Basta probar que si $\alpha_1, \ldots, \alpha_m$ es una demostración en ARP, podemos construir una demostración en ARP^+ que contenga cada α_i entre sus líneas.[3] Supongamos que ya hemos formado una demostración que contenga las líneas anteriores a α_i. Si α_i es un axioma de ARP, entonces podemos incluirlo sin más en la nueva demostración, pues también es un axioma de ARP^+. Ahora tenemos que considerar cuatro casos, según si α_i se deduce de fórmulas anteriores por cada una de las cuatro reglas de inferencia de ARP.

1. Si $\alpha_i \equiv s_1(t) = s_2(t)$ y se deduce por S_1 de $s_1(x) = s_2(x)$, esta última línea la tenemos ya en nuestra nueva deducción, luego podemos aplicar IG para obtener $\bigwedge x s_1(x) = s_2(x)$ y luego EG para obtener $s_1(t) = s_2(t)$.

[1] Hay una sutileza irrelevante, y es que, por ejemplo, el término $x_0 + x_1$, como término de $\mathcal{L}_{\text{arp}}$ es la cadena de signos $\rho pS0S0\kappa SpSSS0SSS0x0xS0$, de longitud 22, mientras que, visto como término de $\mathcal{L}^+_{\text{arp}}$ es la cadena $+x_0x_1$ de longitud 3, porque en $\mathcal{L}^+_{\text{arp}}$ hay bloques de signos de $\mathcal{L}_{\text{arp}}$ que cuentan como un único signo, pero lo importante es que cada expresión de $\mathcal{L}_{\text{arp}}$ se traduce a una única expresión de $\mathcal{L}^+_{\text{arp}}$ agrupando los signos correspondientes a un mismo funtor o variable.

[2] Una *fórmula abierta* es una fórmula sin cuantificadores.

[3] Técnicamente, construimos un funtor $F(d, i)$ que a cada demostración d en ARP y a cada $i < \ell(d)$ le asigna una demostración de la fórmula d_i en ARP^+.

2. Si $\alpha_i \equiv s(t_1) = s(t_2)$ y se deduce por S_2 de $t_1 = t_2$, contamos con esta línea en la nueva deducción, y sólo hay que aplicar ETI para obtener α_i.

3. Si α_i se deduce por la regla T, es claro que podemos deducirla también usando SI y TI.

4. Supongamos finalmente que α_i se deduce por I_0. Esto significa que en la nueva deducción tenemos ya las líneas $s_1(0) = s_2(0)$, $s_1(Sx) = h(x, s_1(x))$, $s_2(Sx) = h(x, s_2(x))$, y tenemos que deducir $s_1(x) = s_2(x)$. Claramente, podemos obtener la conclusión aplicando el principio de inducción abierta a la fórmula $\alpha(x) \equiv s_1(x) = s_2(x)$. No obstante, detallamos la deducción:

(1)	$\alpha(0)$	Premisa
(2)	$s_1(Sx) = h(x, s_1(x))$	Premisa
(3)	$s_2(Sx) = h(x, s_2(x))$	Premisa
(4)	$s_1(x) = s_2(x)$	Hipótesis
(5)	$h(x, s_1(x)) = h(x, s_2(x))$	ETI 4
(6)	$s_1(Sx) = s_2(Sx)$	SI, TI, 2, 3, 5
(7)	$\alpha(x) \to \alpha(Sx)$	
(8)	$\bigwedge u(\alpha(u) \to \alpha(Su))$	IG 7
(9)	$\alpha(0) \wedge \bigwedge u(\alpha(u) \to \alpha(Su))$	IC 1, 8
(10)	$\alpha(0) \wedge \bigwedge u(\alpha(u) \to \alpha(Su)) \to \alpha(x)$	Axioma
(11)	$\alpha(x)$	MP 9, 10

■

Nota Observemos que, un poco más en general, si en ARP podemos deducir una fórmula α de unas premisas, lo mismo sucede en ARP$^+$. (Basta añadir en la prueba anterior el caso en que α_i es una premisa, en cuyo caso podemos incluirla en la deducción nueva también como premisa.) ■

Observemos ahora que en ARP$^+$ tenemos dos definiciones distintas de los conectores lógicos. Por una parte tenemos los conectores del lenguaje $\mathcal{L}^+_{\text{arp}}$ y, por otra, los definidos aritméticamente en ARP. Vamos a ver que son equivalentes. Para ello necesitamos probar antes el teorema siguiente, que afirma que en ARP$^+$ es posible demostrar ciertas fórmulas que, interpretadas con los conectores definidos aritméticamente, ya sabemos que son teoremas de ARP, pero el *quid* es que aquí nos referimos a las fórmulas formadas con los conectores lógicos de $\mathcal{L}^+_{\text{arp}}$, y necesitamos este resultado para probar que son equivalentes a los definidos aritméticamente.

Teorema 4.3 *Las fórmulas siguientes son teoremas de* ARP$^+$*:*

1. $Sx \neq 0$,
2. $Sx = Sy \to x = y$,
3. $x = 0 \vee x = S(x \dot{-} 1)$,
4. $x + z = y + z \to x = y$,

5. $x + y = 0 \to x = 0 \wedge y = 0$,

6. $xy = 0 \to x = 0 \vee y = 0$.

DEMOSTRACIÓN: 1) y 2) son axiomas de ARP^+. Para probar 3) aplicamos el principio de inducción abierta a la fórmula $\alpha(x) \equiv x = 0 \vee x = S(x \dot{-} 1)$. Obviamente tenemos $\alpha(0)$. Supuesto $\alpha(x)$, observamos que $(Sx) \dot{-} 1 = x$, pues esto es un teorema de ARP, luego también de ARP^+ (y aquí no hay conectores). Por lo tanto, $S((Sx) \dot{-} 1) = Sx$, luego tenemos que $Sx = 0 \vee S(Sx \dot{-} 1) = Sx$, que es $\alpha(Sx)$, y esto completa la inducción.

Para probar 4) razonamos por inducción sobre z. Para $z = 0$ tenemos que $x + 0 = x$, $y + 0 = y$, pues esto es un axioma de ARP, luego también de ARP^+. Por lo tanto, $x + 0 = y + 0 \to x = y$.

Supongamos ahora que $x + Sz = y + Sz$. Por los axiomas que definen el funtor suma, esto implica que $S(x + z) = S(y + z)$ y, por 2), concluimos que $x + z = y + z$, de donde la hipótesis de inducción nos da que $x = y$.

Para probar 5) razonamos por inducción sobre y. Para $y = 0$ usamos el axioma $x + 0 = x$, que nos da que $x + 0 = 0 \to x = 0 \wedge 0 = 0$. Supuesto cierto para y, si $x + Sy = 0$, por la definición del funtor suma será $S(x + y) = 0$, lo cual contradice 1), luego podemos concluir cualquier cosa, como $x = 0 \wedge Sy = 0$.

Para probar 6) razonamos por inducción sobre y. Para $y = 0$ es trivial. Si es cierto para y y se cumple $xSy = 0$, entonces, por la definición del funtor producto, $xy + x = 0$, luego por 5) tenemos que $x = 0$, luego también se cumple $x = 0 \vee Sy = 0$. ∎

Recordemos ahora que en $\mathcal{L}_{\mathrm{arp}}$, luego también en $\mathcal{L}^+_{\mathrm{arp}}$, tenemos definida la fórmula

$$x \leq y \equiv x + (y \dot{-} x) = y.$$

Definición 4.4 Llamaremos semifórmulas Δ_0^* de $\mathcal{L}^+_{\mathrm{arp}}$ a las semifórmulas determinadas por los criterios siguientes:

1. Toda semifórmula atómica es Δ_0^*.

2. Si α y β son semifórmulas Δ_0^*, también lo son[4] $\neg\alpha$ y $\alpha \vee \beta$.

3. Si t es un semitérmino, u es una variable que no esté en t, y $\alpha(u)$ es una semifórmula Δ_0^*, entonces

$$\bigvee u \leq t\, \alpha(u) \equiv \bigvee u(u \leq t \wedge \alpha(u)) \quad \text{y} \quad \bigwedge u \leq t\, \alpha(u) \equiv \bigwedge u(u \leq t \to \alpha(u))$$

son semifórmulas Δ_0^*.

De la segunda condición se deduce que $\alpha \wedge \beta$, $\alpha \to \beta$ y $\alpha \leftrightarrow \beta$ también son semifórmulas Δ_0^*. Las fórmulas Δ_0^* son las semifórmulas Δ_0^* que son fórmulas.

[4]Recordemos que estamos suponiendo que los únicos conectores primitivos de $\mathcal{L}^+_{\mathrm{arp}}$ son $\neg$ y $\vee$, pero si queremos considerar los cinco conectores como primitivos, sólo tenemos que añadir en este punto los que faltan.

A cada fórmula α de $\mathcal{L}_a^+$ de tipo Δ_0^* le asociamos una fórmula α^* de $\mathcal{L}_{\rm arp}$ con las mismas variables libres mediante el criterio siguiente:[5]

1. Si α es una fórmula atómica, $\alpha^* \equiv \alpha$.
2. $(\neg\alpha)^* \equiv \neg\alpha^*$.
3. $(\alpha \vee \beta)^* \equiv \alpha^* \vee \beta^*$.
4. $(\bigvee u \le t\, \alpha(u))^* \equiv \bigvee u \le t\, \alpha^*(u)$.
5. $(\bigwedge u \le t\, \alpha(u))^* \equiv \bigwedge u \le t\, \alpha^*(u)$.

Aquí hay que entender que los signos lógicos de los miembros derechos son los definidos aritméticamente en ARP, mientras que los de los miembros izquierdos son los signos lógicos de $\mathcal{L}_{\rm arp}^+$.

Teorema 4.5 *Si α es una fórmula Δ_0^* de $\mathcal{L}_{\rm arp}^+$, entonces en* ARP^+ *se demuestra que $\alpha \leftrightarrow \alpha^*$. En particular, si α^* es demostrable en* ARP*, entonces α lo es en* ARP^+.

Demostración: Razonamos por inducción[6] sobre la longitud de α. El resultado es trivial si α es una fórmula atómica, pues entonces $\alpha^* \equiv \alpha$.

Para los casos restantes observemos en general que en ARP^+ se demuestra, para toda fórmula ϕ de $\mathcal{L}_{\rm arp}$, la equivalencia $t_\phi = 0 \leftrightarrow \phi$ (con el conector de $\mathcal{L}_{\rm arp}^+$), pues, suponiendo $t_\phi = 0$, en ARP (luego en ARP^+) se prueba ϕ, y viceversa, y esto nos da la equivalencia.

Si $\alpha \equiv \neg\beta$ y suponemos que $\beta \leftrightarrow \beta^*$ es demostrable en ARP^+, entonces

$$(\neg\beta)^* \equiv \neg(\beta^*) \equiv 1 \dot{-} t_{\beta^*} = 0 \leftrightarrow t_{\beta^*} \neq 0 \leftrightarrow \neg(\beta^*) \leftrightarrow \neg\beta.$$

Veamos con detalle las dos equivalencias. Para la primera, si $1 \dot{-} t_{\beta^*} = 0$, en ARP podemos probar que $\neg(t_{\beta^*} = 0)$ (con el negador definido en ARP), pero esto implica (siempre en ARP) que $t_{\beta^*} = S(t_{\beta^*} \dot{-} 1)$, luego el apartado 1) del teorema 4.3 nos da que $t_{\beta^*} \neq 0$ (con el negador de ARP^+).

Recíprocamente, si $t_{\beta^*} \neq 0$, por 3) de 4.3, tenemos que $t_{\beta^*} = S(t_{\beta^*} \dot{-} 1)$ y esto implica en ARP que $\neg(t_{\beta^*} = 0)$, de donde $1 \dot{-} t_{\beta^*} = 0$.

La segunda equivalencia (en la que ambos negadores son los de ARP^+) es la que hemos probado en general más arriba.

[5] Técnicamente, consideramos los conjunto $S_i(\alpha)$ formados por las fórmulas de longitud i que se obtienen de las subsemifórmulas de α sustituyendo sus variables ligadas que estén libres por (una elección de) otras variables libres, y definimos un funtor tal que $F(\alpha, i)$ sea una función de dominio $S_i(\alpha)$ de modo que $F(\alpha, i)(\beta) = \beta^*$.

[6] Técnicamente (véase la nota al pie precedente), lo que hacemos es definir por recursión completa un funtor tal que $F(\alpha, i)$ es una función de dominio $S_i(\alpha)$ de modo que $F(\alpha, i)(\beta)$ sea una demostración de $\beta \leftrightarrow \beta^*$.

Si $\alpha \equiv \beta \vee \gamma$ y en ARP^+ se demuestra que $\beta \leftrightarrow \beta^*$ y $\gamma \leftrightarrow \gamma^*$, entonces, usando la propiedad 6) de 4.3,

$$(\beta \vee \gamma)^* \equiv \beta^* \vee \gamma^* \equiv t_{\beta^*} \cdot t_{\gamma^*} = 0 \leftrightarrow t_{\beta^*} = 0 \vee t_{\gamma^*} = 0 \leftrightarrow \beta^* \vee \gamma^* \leftrightarrow \beta \vee \gamma.$$

Si $\alpha \equiv \bigvee u \le t\, \beta(u)$ y el teorema vale para la fórmula $\beta(x)$, entonces

$$(\bigvee u \le t\, \beta(u))^* \equiv \bigvee u \le t\, \beta^*(u)$$

y ahora observamos que, puesto que en ARP, suponiendo $y \le t$ y $\beta^*(y)$ podemos probar $\bigvee u \le t\, \beta^*(u)$ y, recíprocamente, suponiendo $\bigvee u \le t\, \beta^*(u)$ podemos probar $\mu u \le t\, \beta^*(u) \le t$ y $\beta^*(\mu u \le t\, \beta^*(u))$, esto es también así en ARP^+, lo que se traduce en las implicaciones

$$y \le t \wedge \beta^*(y) \to \bigvee u \le t\, \beta^*(u),$$

$$\bigvee u \le t\, \beta^*(u) \to \mu u \le t\, \beta^*(u) \le t \wedge \beta^*(\mu u \le t\, \beta^*(u)),$$

donde las conjunciones y las implicaciones hay que entenderlas ahora como definidas a partir de los conectores de $\mathcal{L}^+_{\mathrm{arp}}$ y no aritméticamente. Esto a su vez se traduce en que $\bigvee u \le t\, \beta^*(u)$ en el sentido aritmético es equivalente en ARP^+ a la fórmula que llamamos igual, pero definida a partir del particularizador y los conectores de $\mathcal{L}^+_a$. Por lo tanto,

$$(\bigvee u \le t\, \beta(u))^* \leftrightarrow \bigvee u \le t\, \beta^*(u) \leftrightarrow \bigvee u \le t\, \beta^*(u) \leftrightarrow \bigvee u \le t\, \beta(u),$$

donde las dos fórmulas intermedias son las correspondientes al particularizador aritmético y al lógico, respectivamente.

Si $\alpha \equiv \bigwedge u \le t\, \beta(u)$, entonces

$$\alpha^* \equiv \bigwedge u \le t\, \beta(u) \equiv \neg \bigvee u \le t\, \neg \beta^*(u),$$

donde todos los signos son los definidos aritméticamente, pero usando los casos ya probados del negador y el particularizador acotado, concluimos que

$$\alpha^* \equiv \bigwedge u \le t\, \beta(u) \equiv \neg \bigvee u \le t\, \neg \beta^*(u) \leftrightarrow \bigwedge u \le t\, \beta(u).$$

■

Teniendo en cuenta las definiciones de los conectores $\wedge, \to, \leftrightarrow$, es fácil concluir que

$$(\beta \wedge \gamma)^* \leftrightarrow \beta^* \wedge \gamma^*, \quad (\beta \to \gamma)^* \leftrightarrow (\beta^* \to \gamma^*), \quad (\beta \leftrightarrow \gamma)^* \leftrightarrow (\beta^* \leftrightarrow \gamma^*).$$

El teorema anterior muestra que toda fórmula α de tipo Δ^*_0 es equivalente en ARP^+ a la fórmula α^*, que es atómica, lo que nos lleva a la definición siguiente:

Definición 4.6 Llamaremos fórmulas Δ_0 en sentido estricto en $\mathcal{L}^+_{\mathrm{arp}}$ a las fórmulas atómicas, mientras que las fórmulas Δ_0 (en sentido amplio) en $\mathcal{L}^+_{\mathrm{arp}}$ serán las fórmulas equivalentes en ARP^+ a fórmulas atómicas.

En estos términos, hemos probado que todas las fórmulas de tipo Δ_0^* son Δ_0 en sentido amplio, pero las fórmulas Δ_0 en sentido amplio incluyen también fórmulas con cuantificadores no acotados acotables. Por ejemplo, si definimos

$$x \mid y \equiv \bigvee z\, y = xz$$

tenemos un cuantificador no acotado, pero es acotable, ya que esta fórmula es equivalente en ARP^+ a $\bigvee z \le y\, y = xz$, que es Δ_0^*. Por lo tanto, tendríamos una definición Δ_0 en sentido amplio. En general tenemos:

Teorema 4.7 *Si α y β son fórmulas Δ_0 en sentido amplio en $\mathcal{L}_{\mathrm{arp}}^+$ y t es un término, también lo son*

$$\neg\alpha, \quad \alpha \vee \beta, \quad \alpha \to \beta, \quad \alpha \wedge \beta, \quad \alpha \leftrightarrow \beta, \quad \bigwedge u \le t\, \alpha(u), \quad \bigvee u \le t\, \alpha(u),$$

donde hay que entender que $\alpha(u)$ es la semifórmula que resulta de sustituir una variable libre en α por la variable ligada u.

Demostración: Basta considerar fórmulas atómicas $\bar\alpha$ y $\bar\beta$ equivalentes a α y β en ARP^+, respectivamente. Entonces, las fórmulas $\neg\bar\alpha$, $\bar\alpha \vee \bar\beta$, etc. son fórmulas Δ_0^* equivalentes a las del enunciado, luego éstas son a su vez equivalentes a fórmulas atómicas, es decir, son Δ_0 en sentido amplio. ∎

Observemos además que si α es una fórmula Δ_0 en sentido amplio y α^* es una fórmula atómica equivalente, el principio de inducción para α^* equivale al principio de inducción para α, por lo que en la práctica el principio de inducción en ARP^+ es válido para fórmulas Δ_0 cualesquiera.

El teorema 4.5 prueba en particular que los conectores y los cuantificadores acotados de ARP^+ son equivalentes a los definidos aritméticamente (pues pasar de α a α^* no es más que sustituir unos por otros y el resultado es una fórmula equivalente), por lo que todos los teoremas que conocemos de ARP son válidos en ARP^+ cuando interpretamos sus conectores y cuantificadores como los de ARP^+.

En [CS 1.20] demostramos el recíproco del teorema 4.5: si una fórmula α de tipo Δ_0^* es demostrable en ARP^+, entonces α^* es demostrable en ARP. En particular, si α es una fórmula de ARP, se trata de una fórmula atómica de ARP^+, luego es Δ_0 y cumple $\alpha \equiv \alpha^*$, lo que nos da el teorema siguiente:

Teorema 4.8 (CS 1.21) *Si una fórmula de $\mathcal{L}_{\mathrm{arp}}$ es demostrable en* ARP^+, *de hecho puede demostrarse en* ARP.

Podemos expresar esto diciendo que ARP^+ es una extensión conservativa de ARP, aunque la situación no encaja exactamente con la definición que dimos en el capítulo anterior porque ARP no es una teoría axiomática de primer orden como las que considerábamos allí, pero la idea es la misma.

En particular, la consistencia de ARP^+ es equivalente a la de ARP.

En la práctica esto significa que la mejor forma de demostrar teoremas de ARP es trabajar en ARP^+, pues cualquier teorema que probemos en ARP^+, si se expresa mediante un fórmula expresable en $\mathcal{L}_{\mathrm{arp}}$ (es decir, mediante una fórmula Δ_0 en sentido amplio), automáticamente podemos asegurar que es un teorema de ARP (y el argumento es constructivo, es decir, podemos diseñar un algoritmo que traduzca la demostración de una fórmula Δ_0 en ARP^+ a demostraciones en ARP de la fórmula equivalente en $\mathcal{L}_{\mathrm{arp}}$). Por lo tanto, sería tonto trabajar con las limitaciones de ARP cuando podemos hacerlo en ARP^+.

La prueba del teorema 4.5 y la nota posterior se traducen en que todos los resultados que hemos demostrado en ARP pueden reinterpretarse ahora como teoremas de ARP^+ considerando a los signos lógicos como los signos lógicos de $\mathcal{L}_a^+$, y no como los definidos aritméticamente.

Nota En lo sucesivo llamaremos ARP a la teoría que hasta aquí hemos venido llamando ARP^+ y llamaremos $\mathcal{L}_{\mathrm{arp}}$ al lenguaje de primer orden que hasta ahora hemos llamado $\mathcal{L}_{\mathrm{arp}}^+$. ∎

A la hora de aplicar en la aritmética recursiva primitiva con cuantificadores los resultados que hemos demostrado sin ellos debemos tener la precaución de que los resultados que habíamos demostrado para fórmulas arbitrarias ahora sólo son aplicables en principio a fórmulas Δ_0.

Por ejemplo, si α es una fórmula Δ_0, podemos considerar el funtor $\chi_\alpha \equiv \chi_{\alpha^*}$ que cumple

$$\chi_\alpha(x_1,\dots,x_n) = 1 \leftrightarrow \alpha(x_1,\dots,x_n), \quad \chi_\alpha(x_1,\dots,x_n) = 0 \leftrightarrow \neg\alpha(x_1,\dots,x_n),$$

pero es necesario que la fórmula sea Δ_0. Similarmente, para que la expresión

$$\mu u \le x\, \alpha(x_1,\dots,x_n,u)$$

defina un funtor necesitamos que la fórmula α sea de tipo Δ_0.

Por otra parte, en la versión de primer orden de ARP ya podemos definir

$$\Pi \underset{T}{\vdash} \alpha \equiv \bigvee d\ \Pi \overset{d}{\underset{T}{\vdash}} \alpha$$

y no necesitamos explicitar las demostraciones en todos los enunciados.

4.2 La aritmética con inducción abierta

En esta sección estudiamos una teoría aritmética más débil que ARP, pero en la sección siguiente veremos que una ligera modificación la convierte en una teoría más fuerte.

Definición 4.9 Definimos el *lenguaje formal de la aritmética de primer orden* como el lenguaje $\mathcal{L}_a$ con igualador y descriptor que tiene por signos eventuales la constante 0, el funtor monádico S (aunque es costumbre escribir $t' \equiv St$) y los funtores diádicos $+$ y $\cdot$.

En $\mathcal{L}_a$ no tenemos el funtor resta truncada, pero a cambio tenemos cuantificadores, luego podemos definir

$$x \leq y \equiv \bigvee u\ u + x = y.$$

Llamaremos semifórmulas abiertas (resp. Δ_0) de $\mathcal{L}_a$ a las determinadas por las condiciones siguientes:

1. Si t_1, t_2 son semitérminos sin descriptores, entonces $t_1 = t_2$ y $t_1 \leq t_2$ son semifórmulas abiertas (y Δ_0).

2. Si α y β son semifórmulas abiertas (resp. Δ_0), también lo son $\neg\alpha$ y $\alpha \vee \beta$.

3. Si t es un semitérmino sin descriptores y α es una semifórmula Δ_0, también lo son $\bigwedge u \leq t\, \alpha$ y $\bigvee u \leq t\, \alpha$ (donde la variable u no está en t).

Aquí estamos adoptando como convenio las abreviaturas:

$$\bigwedge u \leq t\, \alpha \equiv \bigwedge u(u \leq t \to \alpha), \qquad \bigvee u \leq t\, \alpha \equiv \bigvee u(u \leq t \wedge \alpha).$$

Una fórmula es abierta o Δ_0 si es una semifórmula del tipo correspondiente (y además es una fórmula).

Más llanamente, llamamos fórmulas abiertas a las fórmulas sin descriptores que no tienen más cuantificadores que los que aparecen en la definición de $\leq$, y las fórmulas Δ_0 son las que además admiten lo que llamamos *cuantificadores acotados.*

De este modo, si α y β son semifórmulas abiertas (resp. Δ_0) también lo son $\neg\alpha$, $\alpha \vee \beta$, $\alpha \to \beta$, $\alpha \wedge \beta$ y $\alpha \leftrightarrow \beta$ (y en el caso Δ_0 también $\bigwedge u \leq t\, \mathtt{S}^u_x\alpha$ y $\bigvee u \leq t\, \mathtt{S}^u_x\alpha$, donde el término t no tiene descriptores).

La *aritmética con inducción abierta* (resp. Δ_0) es la teoría IA (resp. IΔ_0) sobre el lenguaje $\mathcal{L}_a$ cuyos axiomas son los siguientes:

$$\begin{array}{ll}
(\mathrm{Q1}) & x' \neq 0 \\
(\mathrm{Q2}) & x' = y' \to x = y \\
(\mathrm{Q3}) & x \neq 0 \to \bigvee u\ x = u' \\
(\mathrm{Q4}) & x + 0 = x \\
(\mathrm{Q5}) & x + y' = (x + y)' \\
(\mathrm{Q6}) & x \cdot 0 = 0 \\
(\mathrm{Q7}) & x \cdot y' = xy + x
\end{array}$$

más el *principio de inducción abierta* (resp. Δ_0):

$$\alpha(0) \wedge \bigwedge u(\alpha(u) \to \alpha(u')) \to \alpha(x),$$

para toda fórmula abierta (resp. Δ_0) $\alpha(x)$ de $\mathcal{L}_a$.

Llamando $1 \equiv 0'$, los axiomas Q4 y Q5 implican que

$$x + 1 = x + 0' = (x + 0)' = x'.$$

Por ello, de aquí en adelante escribiremos $x + 1$ en lugar de x'. En estos términos, los axiomas de IA se expresan así:

$$\begin{aligned}
&\text{(Q1)} \quad x + 1 \neq 0\\
&\text{(Q2)} \quad x + 1 = y + 1 \to x = y\\
&\text{(Q3)} \quad x \neq 0 \to \bigvee u\ x = u + 1\\
&\text{(Q4)} \quad x + 0 = x\\
&\text{(Q5)} \quad x + (y + 1) = (x + y) + 1\\
&\text{(Q6)} \quad x \cdot 0 = 0\\
&\text{(Q7)} \quad x \cdot (y + 1) = xy + x\\
&\text{(IA)} \quad \alpha(0) \wedge \bigwedge u(\alpha(u) \to \alpha(u + 1)) \to \alpha(x),
\end{aligned}$$

donde $\alpha(x)$ es una fórmula abierta.

Tomamos también como axioma de IA la fórmula $v|(v = v) = 0$, lo que significa que todas las descripciones impropias serán 0 por definición. No incluimos este axioma en la lista porque en la práctica podemos considerarlo como parte de los axiomas lógicos de IA. Como los axiomas de IA no tienen descriptores, podemos considerar la teoría con los mismos axiomas sobre el lenguaje que resulta de quitarle a $\mathcal{L}_a$ el descriptor, y el teorema 3.29 nos da que IA es una extensión intrascendente de dicha teoría. En el fondo, el axioma $v|(v = v) = 0$ es más un convenio de notación que un axioma.

Podemos identificar los signos de $\mathcal{L}_a$ salvo el descriptor con los signos correspondientes de $\mathcal{L}_{\text{arp}}$, de modo que podemos considerar a toda expresión de $\mathcal{L}_a$ sin descriptores como expresión de $\mathcal{L}_{\text{arp}}$. Más aún, es claro que todos los axiomas de IA son teoremas de ARP, luego todos los teoremas de IA sin descriptores también lo son. (Aquí usamos el teorema 3.29, que nos asegura que los teoremas sin descriptores pueden demostrarse sin descriptores.)

Todos los resultados de esta sección se demuestran en IA.[7]

El teorema siguiente demuestra entre otras cosas que la suma y el producto tienen las propiedades asociativa y conmutativa, por lo que en lo sucesivo escribiremos expresiones de la forma $x_1 + \cdots + x_n$ y $x_1 \cdots x_n$ sin preocuparnos del orden o de la forma en que se asocian los sumandos o factores.[8]

Teorema 4.10 *Se cumple:*

1. $x + (y + z) = (x + y) + z$,

[7] Es decir, que en ARP se demuestra que todos los resultados de esta sección se pueden demostrar en IA.

[8] Si consideramos toda esta exposición formalizada en ARP, aquí estamos apelando a los resultados del último apartado de la sección 2.2 sobre sumas finitas.

2. $x + y = y + x$,

3. $xy = yx$,

4. $(x + y)z = xz + yz$,

5. $x(yz) = (xy)z$,

6. $x + z = y + z \to x = y$,

7. $x + y = 0 \to x = 0 \wedge y = 0$,

8. $xy = 0 \to x = 0 \vee y = 0$.

DEMOSTRACIÓN: 1) Por inducción con[9] $\phi(z) \equiv x + (y + z) = (x + y) + z$. Para $z = 0$ es inmediato y, si vale para z, entonces

$$x + (y + (z + 1)) = x + ((y + z) + 1) = (x + (y + z)) + 1$$

$$= ((x + y) + z) + 1 = (x + y) + (z + 1).$$

2) En primer lugar demostramos que $\bigwedge x(0 + x = x)$. Para ello aplicamos el esquema de inducción a la fórmula $\phi(x) \equiv 0 + x = x$. Ciertamente se cumple para 0 y, si $0 + x = x$, entonces

$$0 + (x + 1) = (0 + x) + 1 = x + 1.$$

Ahora razonamos por inducción con la fórmula $\phi(y) \equiv (x+1)+y = (x+y)+1$. Para $y = 0$ se reduce a $x + 1 = x + 1$. Si se cumple para y tenemos que

$$(x + 1) + (y + 1) = ((x + 1) + y) + 1 = ((x + y) + 1) + 1 = (x + (y + 1)) + 1.$$

Por último consideramos $\phi(y) \equiv x + y = y + x$. Para $y = 0$ es lo primero que hemos probado. Si se cumple $x + y = y + x$, entonces

$$x + (y + 1) = (x + y) + 1 = (y + x) + 1 = (y + 1) + x,$$

por el resultado precedente.

3) Primero aplicamos inducción a la fórmula $\phi(x) \equiv 0 \cdot x = 0$, seguidamente a $\phi(y) \equiv (x + 1)y = xy + y$. Para 0 la comprobación es trivial y

$$(x + 1)(y + 1) = (x + 1)y + x + 1 = xy + y + x + 1 = xy + x + y + 1$$

$$= x(y + 1) + (y + 1).$$

Por último usamos $\phi(x) \equiv xy = yx$. Para 0 es trivial y

$$(x + 1)y = xy + y = yx + y = y(x + 1).$$

[9]Es inmediato (pero debe ser comprobado) que todas las fórmulas a las que aplicamos el principio de inducción son abiertas.

4) Por inducción con $\phi(z) \equiv (x+y)z = xz + yz$

5) Por inducción con $\phi(z) \equiv x(yz) = (xy)z$.

6) Por inducción con $\phi(z) \equiv (x + z = y + z \to x = y)$.

7) Si $y \neq 0$ existe un z tal que $y = z + 1$, luego

$$x + y = x + (z + 1) = (x + z) + 1 \neq 0.$$

Por lo tanto $y = 0$, y esto implica a su vez que $x = 0$.

8) Si $x \neq 0 \wedge y \neq 0$, existen u, v tales que $x = u + 1$, $y = v + 1$, luego

$$xy = x(v + 1) = xv + x = xv + (u + 1) = (xv + u) + 1 \neq 0.$$

■

Nos ocupamos ahora de la relación de orden de los números naturales:

Teorema 4.11 *Se cumple:*

1. $x \leq x$
2. $x \leq y \wedge y \leq x \to x = y$
3. $x \leq y \wedge y \leq z \to x \leq z$
4. $0 \leq x$
5. $x \leq y \vee y \leq x$
6. $x \leq x + 1$
7. $x \leq y + 1 \to x \leq y \vee x = y + 1$
8. $x \leq y \leftrightarrow x + z \leq y + z$
9. $z \neq 0 \wedge xz = yz \to x = y$
10. $z \neq 0 \to (x \leq y \leftrightarrow xz \leq yz)$

DEMOSTRACIÓN: 1) es trivial, pues $x + 0 = x$.

2) Tenemos que $u + x = y$ y $v + y = x$, luego $u + v + y = 0 + y$, luego $u + v = 0$, luego $u = 0$, luego $x = y$.

3) Tenemos que $x + u = y$ y $y + v = z$, luego $x + u + v = z$, luego $x \leq z$.

4) Como $x + 0 = x$, tenemos que $0 \leq x$.

5) Por inducción con $\phi(x) \equiv (x \leq y \vee y \leq x)$.

Para $x = 0$ tenemos que $0 \leq y$. Supuesto cierto para x, si $y \leq x$, entonces existe un z tal que $y + z = x$, luego $y + z + 1 = x + 1$, luego $y \leq x + 1$. Si $x \leq y$, entonces $x + z = y$. Si $z = 0$ tenemos que $y = x \leq x + 1$, y si $z \neq 0$ entonces $z = u + 1$, luego $u + x + 1 = y$, luego $x + 1 \leq y$.

6) es consecuencia inmediata de que $x + 1 = x + 1$.

7) Tenemos que $x + u = y + 1$. Distinguimos dos casos: si $u = 0$, entonces $x = y+1$, y en caso contrario existe un v tal que $u = v+1$, luego $x+v+1 = y+1$, luego $x + v = y$, luego $x \leq y$.

8) Se cumple $x \leq y$ si y sólo si existe un u tal que $x + u = y$, si y sólo si $x + z + u = y + z$ si y sólo si $x + z \leq y + z$.

9) Si $xz = yz$, no perdemos generalidad si suponemos $x \leq y$, es decir, $x + u = y$. Entonces $xz + uz = yz$, luego $uz = 0$, luego $u = 0$, luego $x = y$.

10) Si $x \leq y$, entonces $x + u = y$, luego $xz + uz = yz$, luego $xz \leq yz$. Si $xz \leq yz$, o bien $x \leq y$, como queremos probar, o bien $y \leq x$, en cuyo caso $yz \leq xz$, luego $yz = xz$, luego $x = y$, luego $x \leq y$. ∎

Los primeros apartados del teorema anterior se resumen diciendo que $\leq$ es una relación de orden total cuyo mínimo es 0 y respecto a la que $x + 1$ es el menor número natural mayor que x. Escribiremos

$$x < y \equiv x \leq y \wedge x \neq y \leftrightarrow \bigvee u(u \neq 0 \wedge x + u = y).$$

Es inmediato que

$$x < y \leftrightarrow x + z < y + z, \qquad z \neq 0 \rightarrow (x < y \leftrightarrow xz < yz).$$

Veamos ahora un primer ejemplo de la utilidad del descriptor. Si $x \leq y$, tenemos que existe un z tal que $x + z = y$, y por 4.10 6) este z resulta estar unívocamente determinado. Esto justifica la definición siguiente:

Definición 4.12 $y \dot{-} x \equiv z \mid (y = z + x)$.

Según acabamos de observar, $y \dot{-} x$ es una descripción propia si y sólo si $x \leq y$, pero, por el convenio que hemos adoptado sobre las descripciones impropias tenemos que $y < x \rightarrow y \dot{-} x = 0$.

El teorema siguiente se demuestra con facilidad:

Teorema 4.13 *Se cumple*

1. $\bigwedge xyz(x \geq y \rightarrow x \dot{-} y = (x + z) \dot{-} (y + z))$,
2. $\bigwedge xyz(x \geq y \rightarrow (x \dot{-} y) + z = (x + z) \dot{-} y)$,
3. $\bigwedge xyz\ (x \dot{-} y)z = xz \dot{-} yz$.

Ahora demostramos el teorema sobre la división euclídea:

Teorema 4.14 $\bigwedge xy(y \neq 0 \rightarrow \overset{1}{\bigvee} cr(r < y \wedge x = yc + r)$

DEMOSTRACIÓN: Sea $\phi(c) \equiv yc \leq x$. Se cumple $\phi(0)$ y existe un c tal que $\neg\phi(c)$, por ejemplo, $c = x + 1$, pues $y(x + 1) \geq 1(x + 1) > x$. Por lo tanto, no puede ser cierto $\bigwedge c(\phi(c) \to \phi(c+1))$, ya que entonces por inducción tendríamos $\bigwedge c\, \phi(c)$. Así pues, existe un c tal que $\phi(c) \wedge \neg\phi(c + 1)$. Explícitamente, esto significa que $yc \leq x < yc + y$. En particular $c \leq yc \leq x$. Sea $r = x \dot{-} yc$, de modo que $x = yc + r < yc + y$, luego $r < y$.

Para probar la unicidad suponemos $x = yc + r = yc' + r'$ con $r, r' < y$. No perdemos generalidad si suponemos $r \leq r'$. Entonces $yc = yc' + (r' \dot{-} r)$, luego $yc' \leq yc$, luego $y(c \dot{-} c') = yc \dot{-} yc' = r' \dot{-} r \leq r' < y$. Si $r \neq r'$ entonces tiene que ser $c = c'$, porque en caso contrario $y \leq y(c \dot{-} c') < y$. Pero entonces $yc + r = yc + r'$ y $r = r'$. Concluimos que $r = r'$, y entonces $yc = yc'$, luego $c = c'$. ∎

Necesitaremos también las propiedades básicas de la divisibilidad:

Definición 4.15 $x \mid y \equiv \bigvee u\ y = xu$.

Teorema 4.16 *Se cumple:*

1. $1 \mid x \wedge x \mid x \wedge x \mid 0$,
2. $x \mid y \wedge y \neq 0 \to x \leq y$,
3. $x \mid y \wedge y \mid z \to x \mid z$,
4. $x \mid y \wedge y \mid x \to x = y$,
5. $x \mid y \to x \mid yz$,
6. $x \mid y \wedge x \mid z \to x \mid (y + z)$.

DEMOSTRACIÓN: 1) es inmediato. Para probar 2) observamos que si $y = xz$ con $y \neq 0$, entonces $z \neq 0$, luego $1 \leq z$, luego $x \leq xz = y$.

Para 4) observamos que si $y = ux \wedge x = vy$ entonces, descartando el caso trivial en que $x = y = 0$, tenemos que $y = uvy$, luego $uv = 1$ y esto implica $u = 1$ por 2). Todo lo demás es sencillo. ∎

Con esto estamos en condiciones de definir en IA los pares ordenados $\langle x, y\rangle_2$ que en 2.3 definimos para ARP.

Diremos que un número x es *par* si $2 \mid x$, y en caso contrario es *impar*. Notemos que $2 \mid x \vee 2 \mid x + 1$. Basta expresar $x = 2u + r$, con $r = 0, 1$.

En particular, o bien $2 \mid (x + y)$ o bien $2 \mid (x + y + 1)$, luego en cualquier caso $2 \mid (x + y)(x + y + 1)$, y también $2 \mid (x + y)(x + y + 1) + 2x$, luego existe un (único) z tal que

$$2z = (x + y)(x + y + 1) + 2x.$$

Esto justifica la definición siguiente (compárese con 2.3):

Definición 4.17 $\langle x, y\rangle \equiv z \mid 2z = (x + y)(x + y + 1) + 2x$.

Teorema 4.18 $\bigwedge z \overset{1}{\bigvee} xy\ z = \langle x, y\rangle$.

DEMOSTRACIÓN: Sea $\phi(r) \equiv r(r+1) \leq 2z$. Claramente $\phi(0) \wedge \neg\phi(z+1)$, luego no puede realizarse la inducción sobre ϕ, es decir, existe un r tal que $\phi(r) \wedge \neg\phi(r+1)$. Explícitamente, $r(r+1) \leq 2z < (r+1)(r+2)$. Es fácil ver que r es único, pues si $r' \leq r$ se cumple $\phi(r')$ y si $r' > r$ se cumple $\neg\phi(r')$.

Como $2z - r(r+1)$ es par, existe un x tal que $2x = 2z - r(r+1)$. Se cumple que $x \leq r$, pues en caso contrario $2r < 2z - r^2 - r$, luego $r^2 + 3r < 2z$. Es fácil ver que la suma y el producto de pares es par, que el producto de impares es impar y que la suma de impares es par. De aquí se sigue (distinguiendo casos según que r sea par o impar) que $r^2 + 3r$ es par en cualquier caso, luego $r^2 + 3r + 1 \leq 2z$ no puede cumplirse con igualdad, luego $r^2 + 3r + 2 \leq 2z$, es decir, $(r+1)(r+2) \leq 2z$, contradicción.

Así pues, $x \leq r$ y existe un y tal que $x + y = r$. En definitiva llegamos a que $2z = 2x + (x+y)(x+y+1)$, que es lo mismo que $z = \langle x, y\rangle$.

Si $\langle x, y\rangle = \langle x', y'\rangle$, llamamos $r' = x' + y'$, de modo que $2z = r'(r'+1) + 2x'$, con $x' \leq r'$, luego

$$r'(r'+1) \leq 2z \leq r'^2 + r' + 2r' < r'^2 + 3r' + 2 = (r'+1)(r'+2).$$

Por la unicidad de r resulta que $r' = r$, luego $2x = 2x'$, luego $x = x'$, luego $y = y'$. ∎

Más explícitamente, la unicidad que hemos probado equivale a que

$$\bigwedge xyx'y'(\langle x, y\rangle = \langle x', y'\rangle \to x = x' \wedge y = y').$$

Definición 4.19 $z_1 \equiv x \mid \bigvee y\ z = \langle x, y\rangle, \qquad z_2 \equiv y \mid \bigvee x\ z = \langle x, y\rangle.$

Así, para todo z se cumple que $z = \langle z_1, z_2\rangle$.

Notemos que $x, y \leq \langle x, y\rangle$. En efecto, podemos suponer que $x + y \geq 1$, y entonces

$$2\,\langle x, y\rangle \geq (x+y)(x+y+1) = (x+y)^2 + x + y \geq x + y + x + y = 2x + 2y,$$

luego $\langle x, y\rangle \geq x + y$. ∎

4.3 La jerarquía de Kleene

En general, toda fórmula de una teoría axiomática es lógicamente equivalente a otra en forma prenexa (véase 3.23), y esto nos da una clasificación de las fórmulas según su complejidad. Vamos a ver que en IA esta clasificación es mucho más simple gracias a que, según acabamos de ver, cada número natural determina un par de números naturales.

Definición 4.20 (Jerarquía de Kleene) Para cada número natural $n \geq 1$, diremos que una fórmula de $\mathcal{L}_a$ es de tipo Σ_n (resp. Π_n) si es de la forma[10]

$$\bigvee u_1 \bigwedge u_2 \cdots \alpha \qquad \text{o} \qquad \bigwedge u_1 \bigvee u_2 \cdots \alpha,$$

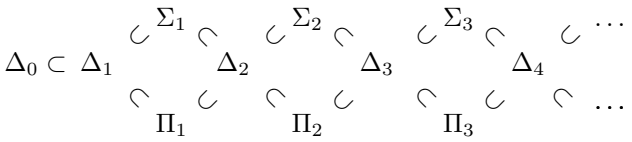
donde α es una semifórmula de tipo Δ_0 y el número de cuantificadores alternados es $\leq n$, pero si es exactamente n, entonces el primero es $\bigvee$ en el caso de las fórmulas Σ_n y es $\bigwedge$ en el caso de las fórmulas Π_n.

Más en general, si T es una teoría axiomática[11] en la que pueden probarse los axiomas de IA, diremos que una fórmula cualquiera es de tipo Σ_n^T o Π_n^T si es equivalente en T a una fórmula del tipo correspondiente. Una fórmula es de tipo Δ_n^T si es a la vez Σ_n^T y Π_n^T. Normalmente omitiremos el superíndice T si está claro por el contexto en qué teoría estamos trabajando.

La idea de fondo es que una fórmula Σ_n pretende ser una fórmula que tenga exactamente n cuantificadores alternados empezando por un particularizador, pero, por una parte, admitimos que tenga menos porque, por ejemplo, $\bigvee u \bigwedge v \alpha$ es una fórmula Σ_2 que también puede verse como Σ_3, ya que si añadimos un cuantificador $\bigvee u \bigwedge v \bigvee w \alpha$, donde w es una variable que no esté en α, obtenemos una fórmula Σ_3 equivalente. Por otro lado, también admitimos $\bigwedge u \bigvee v \alpha$ como fórmula de tipo Σ_3, ya que es equivalente a $\bigvee w \bigwedge u \bigvee v \alpha$. Vemos así que, en teoría, siempre podemos suponer que una fórmula de tipo Σ_n empieza por un particularizador, mientras que una de tipo Π_n empieza por un generalizador.

Igualmente es obvio que toda fórmula de tipo Σ_n o Π_n es tanto Σ_{n+1} como Π_{n+1}, luego ambas son Δ_{n+1}^T.

Por lo tanto, entre las clases de fórmulas de la jerarquía de Kleene se dan las inclusiones siguientes:



Observemos que la jerarquía de Kleene puede definirse también para fórmulas de $\mathcal{L}_{\mathrm{arp}}$, partiendo en este caso de las fórmulas atómicas como fórmulas Δ_0. Por ejemplo, el teorema siguiente también es válido cambiando IA por ARP. Sólo hay que tener en cuenta que en ARP también tenemos definidos los pares ordenados $\langle x, y\rangle$, de modo que la fórmula $z = \langle x, y\rangle$ es atómica, luego Δ_0.

[10] Técnicamente, es fácil definir un funtor tal que si α es una fórmula de $\mathcal{L}_a$, entonces $F(\alpha) = \langle \pi, \beta\rangle$, donde π es una sucesión de cuantificadores y variables de la mayor longitud posible tal que $\alpha = \pi^\frown\beta$. Esto nos permite definir funtores diádicos Σ_n y Π_n de modo que $\Sigma_n(\alpha) = 1$ (resp. $\Pi_n(\alpha) = 1$) si y sólo si α es una fórmula de tipo Σ_n (resp. Π_n). A su vez, podemos considerar las fórmulas Δ_0 dadas por $\alpha \in \Sigma_n \equiv \Sigma_n(\alpha) = 1$, $\alpha \in \Pi_n \equiv \Pi_n(\alpha) = 1$.

[11] Admitimos que el lenguaje de T tenga más signos aparte de los de $\mathcal{L}_a$, pero en tal caso hay que tener presente que las fórmulas de tipo Δ_0 en sentido estricto sólo pueden tener los signos de $\mathcal{L}_a$, sin perjuicio de que consideremos también fórmulas de tipo Δ_0 a las que sean equivalentes en T a fórmulas Δ_0 en sentido estricto, aunque contengan más signos.

Teorema 4.21 *Sea T una teoría axiomática que extienda a* IA. *Para cada número natural $n \geq 1$ y para fórmulas α y β cualesquiera se cumple:*[12]

1. *Si α, β son Σ_n, Π_n, lo mismo vale para $\alpha \wedge \beta$ y $\alpha \vee \beta$.*
2. *Si α es Π_n (resp. Σ_n) y β es Σ_n (resp. Π_n), $\alpha \to \beta$ es Σ_n (resp. Π_n).*
3. *Si α es Σ_n entonces $\neg\alpha$ es Π_n, y viceversa.*
4. *Si $\alpha(x)$ es Σ_n, también lo es $\bigvee u\,\alpha(u)$.*
5. *Si $\alpha(x)$ es Π_n, también lo es $\bigwedge u\,\alpha(u)$.*

DEMOSTRACIÓN: Veamos primero 4). Consideremos una fórmula α de clase Σ_n. Esto significa que es equivalente a otra de la forma $\bigvee y\,\pi\,\beta$, donde π es una sucesión de $n-1$ cuantificadores alternados (tal vez ninguno, pero, si los hay, el primero es un generalizador) y β es Δ_0. Entonces $\bigvee x\alpha$ es equivalente a $\bigvee xy\,\pi\,\beta$, y basta probar que esta fórmula es Σ_n. Ahora bien:

$$\bigvee xy\,\pi\,\beta \leftrightarrow \bigvee u\bigwedge xy(u = \langle x, y\rangle \to \pi\,\beta) \leftrightarrow \bigvee u\bigwedge x \leq u\bigwedge y \leq u(u = \langle x, y\rangle \to \pi\,\beta)$$
$$\leftrightarrow \bigvee u\pi\,\bigwedge x \leq u\bigwedge y \leq u(u = \langle x, y\rangle \to \beta),$$

donde hemos extraído π usando el teorema 3.24. La fórmula tras $\bigvee u\,\pi$ es Δ_0, luego la fórmula completa es Σ_n.

La prueba de 5) es análoga, considerando las equivalencias

$$\bigwedge xy\,\pi\,\beta \leftrightarrow \bigwedge u\bigvee xy(u = \langle x, y\rangle \wedge \pi\,\beta) \leftrightarrow \bigwedge u\bigvee x \leq u\bigvee y \leq u(u = \langle x, y\rangle \wedge \pi\,\beta)$$
$$\leftrightarrow \bigwedge u\,\pi\bigvee x \leq u\bigvee y \leq u(u = \langle x, y\rangle \wedge \beta).$$

Para probar 1) tomemos fórmulas α y β de tipo Σ_n, es decir, equivalentes a fórmulas

$$\bigvee u_0\bigwedge u_1 \cdots \alpha', \qquad \bigvee v_0\bigwedge v_1 \cdots \beta',$$

donde α' y β' son de tipo Δ_0. Podemos suponer que las variables u_i y v_i son todas distintas entre sí. Entonces, usando de nuevo 3.24, vemos que $\alpha \wedge \beta$ es equivalente a

$$\bigvee u_1v_1\bigwedge u_2v_2 \cdots (\alpha' \wedge \beta'),$$

y aplicando 4) y 5) concluimos que esta fórmula es de tipo Σ_n. El caso de $\alpha \vee \beta$ es idéntico. Por último, 3) es inmediata, pues al anteponer un negador a un prefijo de cuantificadores alternados podemos pasarlo a la derecha del prefijo invirtiendo cada cuantificador y 2) es consecuencia de 1) y 3), ya que $\alpha \to \beta$ equivale a $\neg\alpha \vee \beta$. ∎

Notemos que toda fórmula del lenguaje $\mathcal{L}_a$ es de tipo Σ_n o Π_n para algún n, pues las fórmulas atómicas son Δ_0, y el teorema anterior asigna un puesto en la jerarquía a todas las fórmulas que vamos construyendo a partir de ellas.[13]

[12] Véase además el teorema 4.26, más abajo.

[13] No importa si no encajan exactamente en las hipótesis. Por ejemplo, si tenemos una fórmula de tipo $\Sigma_3 \to \Sigma_3$, también podemos considerar que es $\Pi_4 \to \Sigma_4$, y el resultado es Σ_4 (de hecho Δ_4).

Para determinar la posición en la jerarquía de Kleene de una fórmula con descriptores es útil la definición siguiente:

Definición 4.22 Diremos que un término t de $\mathcal{L}_a$ (o de cualquier lenguaje que contenga los signos de $\mathcal{L}_a$) es Σ_n, Π_n o Δ_n en una teoría T si y sólo si lo es la fórmula $x = t$, donde x es una variable que no esté en t.

Notemos que si un término t es Σ_n (en una teoría que extienda a IA) entonces es Δ_n, pues

$$x = t \leftrightarrow \bigwedge u(u = t \to x = u).$$

Por otra parte, si $\phi(x_1, \ldots, x_n)$ es una fórmula Σ_n, Π_n o Δ_n y $t_1, \ldots, t_n$ son términos Σ_n (en una teoría que extienda a IA), entonces la fórmula $\phi(t_1, \ldots, t_n)$ es del mismo tipo que ϕ. En efecto:

$$\phi(t_1, \ldots, t_n) \leftrightarrow \bigvee x_1 \cdots x_n(x_1 = t_1 \wedge \cdots \wedge x_n = t_n \wedge \phi(x_1, \ldots, x_n))$$

$$\leftrightarrow \bigwedge x_1 \cdots x_n(x_1 = t_1 \wedge \cdots \wedge x_n = t_n \to \phi(x_1, \ldots, x_n)).$$

Definición 4.23 Llamaremos $\mathrm{I}\Sigma_n$ a la teoría que resulta de extender IA admitiendo el principio de inducción para fórmulas de tipo Σ_n en lugar de únicamente para fórmulas abiertas.[14] Si admitimos el principio de inducción para fórmulas aritméticas (es decir, para fórmulas arbitrarias de $\mathcal{L}_a$) obtenemos la llamada *Aritmética de Peano* (de primer orden) AP.

Una vez que contamos con el principio de inducción para fórmulas Σ_1, podemos suprimir el axioma Q3, es decir, $\alpha(x) \equiv x \neq 0 \to \bigvee u\ x = u + 1$, ya que éste puede probarse por inducción, pues tanto $\alpha(0)$ como $\alpha(x+1)$ son triviales.

Así, podemos tomar como axiomas de AP los siguientes (más $v|(v = v) = 0$):

(AP1) $x + 1 \neq 0$
(AP2) $x + 1 = y + 1 \to x = y$
(AP3) $x + 0 = x$
(AP4) $x + (y + 1) = (x + y) + 1$
(AP5) $x \cdot 0 = 0$
(AP6) $x(y + 1) = xy + x$
(AP7) $\alpha(0) \wedge \bigwedge u(\alpha(u) \to \alpha(Su)) \to \alpha(x)$,

para toda fórmula α de $\mathcal{L}_a$. Si restringimos el principio de inducción a fórmulas α de tipo Σ_n, tenemos los axiomas de $\mathrm{I}\Sigma_n$.

[14] Sólo son axiomas las fórmulas de inducción construidas con fórmulas Σ_n en sentido estricto, pero es claro que cualquier principio de inducción sobre una fórmula Σ_n es equivalente al principio que resulta de sustituirla por la fórmula Σ_n en sentido estricto equivalente, así que en la práctica no hay diferencia.

En principio, podemos definir $\mathrm{I}\Pi_n$ como la teoría que resulta de extender IA con el principio de inducción para fórmulas de tipo Π_n, pero el teorema siguiente demuestra que $\mathrm{I}\Sigma_n$ e $\mathrm{I}\Pi_n$ son la misma teoría.

Teorema 4.24 *En $\mathrm{I}\Sigma_n$ se puede probar el principio de inducción para fórmulas de tipo Π_n, mientras que en $\mathrm{I}\Pi_n$ se puede probar el principio de inducción para fórmulas de tipo Σ_n.*

DEMOSTRACIÓN: Tomemos una fórmula ϕ de tipo Π_n y supongamos

$$\phi(0) \wedge \bigwedge u(\phi(u) \to \phi(u+1)).$$

Queremos probar que $\phi(x)$. Supongamos, por reducción al absurdo, que existe un a tal que $\neg\phi(a)$. Aplicamos inducción a la fórmula

$$\psi(z) \equiv z \le a \to \bigvee u(u + z = a \wedge \neg\phi(u)),$$

que es de tipo Σ_n (en $\mathrm{I}\Sigma_n$). Obviamente se cumple $\psi(0)$ y, supuesto $\psi(z)$, si $z+1 \le a$, entonces $z \le a$, luego por $\psi(z)$ existe un u tal que $u+z = a \wedge \neg\phi(u)$. No puede ser $u = 0$, luego $u = v+1$, con $z+1+v = a$, y tiene que ser $\neg\phi(v)$, pues $\phi(v) \to \phi(u)$. Así pues, $\bigvee v(v + z + 1 = a \wedge \neg\phi(v))$, y esto es $\psi(z+1)$.

Por Σ_n-inducción tenemos $\bigwedge x\, \psi(x)$. En particular $\psi(a)$, que implica $\neg\phi(0)$, contradicción.

El recíproco se prueba análogamente: si partimos de una fórmula $\phi(x)$ de tipo Σ_n y suponemos que cumple las hipótesis del principio de inducción pero existe un a tal que $\neg\phi(a)$, aplicamos Π_n-inducción a la fórmula

$$\psi(z) \equiv \bigwedge u(z + u = a \to \neg\phi(u)).$$

El argumento es muy similar. ∎

Claramente tenemos $\mathrm{IA} \subset \mathrm{I}\Sigma_1 \subset \mathrm{I}\Sigma_2 \subset \mathrm{I}\Sigma_3 \subset \cdots \mathrm{AP}$, donde la inclusión entre dos teorías quiere decir que la segunda extiende a la primera. Notemos que todo teorema de AP es demostrable en $\mathrm{I}\Sigma_n$ para un n suficientemente grande, pues la prueba sólo puede usar un número finito de instancias de inducción y las fórmulas correspondientes serán Σ_n para un n suficientemente grande. Sin embargo, la mayor parte de los resultados que vamos a presentar pueden probarse de hecho en $\mathrm{I}\Sigma_1$.

Un resultado muy importante sobre la jerarquía de fórmulas que hemos definido es que los cuantificadores acotados no aumentan la complejidad de una fórmula. Para probarlo necesitamos un hecho previo:

Teorema 4.25 (Principio de Recolección) *Para toda fórmula $\phi(x,y)$ (posiblemente con más variables libres) la fórmula siguiente es un teorema de* AP:

$$\bigwedge u \le x \bigvee v\, \phi(u,v) \to \bigvee w \bigwedge u \le x \bigvee v \le w\, \phi(u,v).$$

DEMOSTRACIÓN: Supongamos $\bigwedge u \le x \bigvee v\, \phi(u,v)$. Vamos a aplicar el principio de inducción a la fórmula

$$\psi(z) \equiv z \le x+1 \to \bigvee w \bigwedge u < z \bigvee v \le w\, \phi(u,v).$$

Para $z = 0$ es trivial. Si vale para z, suponemos que $z+1 \le x+1$. Por hipótesis de inducción existe un w tal que $\bigwedge u < z \bigvee v \le w\, \phi(u,v)$. Por otra parte, la hipótesis del teorema nos da un y tal que $\phi(z,y)$ y sea w' el máximo de w, y. Entonces

$$\bigwedge u < z+1 \bigvee v \le w'\, \phi(u,v).$$

Concluimos que se cumple $\psi(z+1)$, y eso es lo que queríamos probar. En particular tenemos $\psi(x+1)$, de donde se sigue la conclusión. ∎

Teorema 4.26 *Si α es una fórmula Σ_n, Π_n o Δ_n, también lo son las fórmulas $\bigwedge u \le t\, \alpha$ y $\bigvee u \le t\, \alpha$.*

DEMOSTRACIÓN: Lo probamos por inducción sobre n. Si α es Σ_n es equivalente a una de la forma $\bigvee v\, \phi(u,v)$, con ϕ de tipo Π_{n-1} (entendiendo que Π_0 es Δ_0). El teorema anterior nos da la equivalencia:

$$\bigwedge u \le t \bigvee v\, \phi(u,v) \leftrightarrow \bigvee w \bigwedge u \le t \bigvee v \le w\, \phi(u,v).$$

Si $n = 1$ la fórmula tras el $\bigvee w$ es Δ_0, luego la fórmula completa es Σ_1, como había que probar. Para $n > 1$ la fórmula es Π_{n-1} por hipótesis de inducción, luego la fórmula completa es Σ_n igualmente. La clausura respecto a $\bigvee w$ se sigue de 4.21.

Notemos que el caso $\bigvee u \le t\, \alpha$ es trivial, pues los particularizadores conmutan. Si α es Π_n razonamos igual, pero aplicando la equivalencia a la fórmula $\neg\phi$ y negando ambas partes, con lo cual nos queda que

$$\bigvee u \le t \bigwedge v\, \phi(u,v) \leftrightarrow \bigwedge w \bigvee u \le t \bigwedge v \le w\, \phi(u,v).$$

∎

Ahora, refinando y combinando las pruebas de los dos teoremas anteriores, obtenemos lo siguiente:

Teorema 4.27 *En $\mathrm{I}\Sigma_n$:*

- *Se demuestra el principio de recolección para fórmulas Σ_n,*
- *Las fórmulas Σ_n y Π_n son cerradas para cuantificadores acotados.*

DEMOSTRACIÓN: En efecto, por abreviar llamaremos $R(\Gamma)$ al principio de recolección para fórmulas de tipo Γ y A_n al hecho de que las fórmulas Σ_n y Π_n son cerradas para cuantificadores acotados.

Ahora observamos que si ϕ es Δ_0 en el teorema 4.25, la fórmula ψ sobre la que se aplica la inducción es Σ_1, luego la prueba vale en $\mathrm{I}\Sigma_1$. A su vez, la prueba de 4.26 para fórmulas Σ_1 y Π_1 usa el principio de recolección para fórmulas Δ_0, luego $\mathrm{I}\Sigma_1$ demuestra A_1. Volvemos entonces a la prueba de 4.25 y vemos que, si ϕ es Σ_1, teniendo en cuenta A_1, la fórmula ψ a la que se le aplica la inducción es también Σ_1, luego concluimos que en $\mathrm{I}\Sigma_1$ se prueba $R(\Sigma_1)$.

En general, si suponemos que en $\mathrm{I}\Sigma_n$ se demuestra $R(\Pi_{n-1})$, A_n y $R(\Sigma_n)$ (entendiendo que Π_0 es lo mismo que Δ_0), vemos que si ϕ es Π_n en 4.25, usando A_n, resulta que la inducción es Σ_{n+1}, luego $\mathrm{I}\Sigma_{n+1}$ demuestra $R(\Pi_n)$. A continuación, 4.26, a partir de $R(\Sigma_n)$ y $R(\Pi_n)$ (luego en $\mathrm{I}\Sigma_{n+1}$), demuestra A_{n+1} y, por último, si ϕ es Σ_{n+1}, usando A_{n+1} vemos que la inducción de 4.25 es Σ_{n+1}, con lo que concluimos que $\mathrm{I}\Sigma_{n+1}$ demuestra $R(\Pi_n)$, A_{n+1} y $R(\Sigma_{n+1})$. Por inducción el teorema vale para todo n. ∎

Seguidamente demostramos la variante fuerte del principio de inducción:

Teorema 4.28 *Si $\phi(x)$ es cualquier fórmula, la fórmula siguiente es un teorema de* AP *(y se demuestra en $\mathrm{I}\Sigma_n$ para fórmulas Σ_n o Π_n):*

$$\bigwedge x(\bigwedge y < x\ \phi(y) \to \phi(x)) \to \bigwedge x\,\phi(x).$$

Demostración: Por inducción sobre $\psi(x) \equiv \bigwedge y < x\,\phi(y)$. Para $x = 0$ se cumple trivialmente. Supongamos que $\bigwedge y < x\,\phi(y)$. Entonces por hipótesis $\phi(x)$. Veamos que $\bigwedge y < x+1\,\phi(y)$. En efecto, si $y < x+1$, entonces $y \le x$, luego o bien $y < x$ (en cuyo caso $\phi(y)$ por hipótesis de inducción) o bien $y = x$ (en cuyo caso ya hemos observado que se cumple $\phi(y)$). Concluimos que $\bigwedge x\psi(x)$, luego, para todo x se cumple $\psi(x+1)$, lo cual implica $\phi(x)$. ∎

El teorema siguiente afirma que si existe un número natural que cumple una propiedad entonces existe un mínimo número que la cumple.

Teorema 4.29 *Si $\phi(x)$ es cualquier fórmula, la fórmula siguiente es un teorema de* AP *(y se demuestra en $\mathrm{I}\Sigma_n$ para fórmulas Σ_n o Π_n):*

$$\bigvee x\phi(x) \to \overset{1}{\bigvee} x(\phi(x) \wedge \bigwedge y < x\,\neg\phi(y)).$$

Demostración: Supongamos $\bigvee x\phi(x)$. Si la existencia es falsa, tenemos que $\bigwedge x(\phi(x) \to \bigvee y < x\ \phi(y))$. Ahora razonamos por inducción con la fórmula $\psi(x) \equiv \bigwedge y < x\,\neg\phi(y)$. Obviamente se cumple para 0. Si vale para x, tomemos un $y < x+1$, es decir, $y \le x$. Si $y < x$ se cumple $\neg\phi(y)$ por hipótesis de inducción, mientras que si $y = x$ también tiene que ser $\neg\phi(x)$, ya que en caso contrario tendría que existir un $y < x$ que cumpliera $\phi(y)$, y no existe.

Esto prueba $\bigwedge x\psi(x)$. Por lo tanto, para todo x tenemos $\psi(x+1)$, luego $\neg\phi(x)$, contradicción. La unicidad es clara. ∎

Un enunciado claramente equivalente del teorema anterior es

$$\bigvee x\phi(x) \to \overset{1}{\bigvee} x(\phi(x) \wedge \bigwedge y(\phi(y) \to x \le y)).$$

Bajo la hipótesis obvia de acotación también podemos justificar la existencia de máximo:

Teorema 4.30 *Si $\phi(x)$ es cualquier fórmula, la fórmula siguiente es un teorema de* AP *(y se demuestra en $\mathrm{I}\Sigma_n$ para fórmulas Σ_n o Π_n):*

$$\bigvee x\,\phi(x) \wedge \bigvee y \bigwedge u(\phi(u) \to u \le y) \to \overset{1}{\bigvee} x(\phi(x) \wedge \bigwedge u(\phi(u) \to u \le x)).$$

DEMOSTRACIÓN: Tomemos y según la hipótesis. Basta aplicar el teorema anterior a la fórmula

$$\psi(z) \equiv z \leq y \wedge \phi(y \dot{-} z) \leftrightarrow \bigvee u \leq z(u + z = y \wedge \phi(u)).$$

Si z es el mínimo que que cumple $\psi(z)$, tomamos $x = y - z$ y claramente cumple lo pedido. ∎

Definición 4.31

$$\text{mín}\, x|\phi(x) \equiv x \mid (\phi(x) \wedge \bigwedge y(\phi(y) \to x \leq y)),$$

$$\text{máx}\, x|\phi(x) \equiv x \mid (\phi(x) \wedge \bigwedge y(\phi(y) \to y \leq x)),$$

Los dos teoremas anteriores dan condiciones suficientes (y de hecho necesarias) para que estas descripciones sean propias. Observemos también que los cuatro últimos teoremas se prueban en $\mathrm{I}\Delta_0$ para fórmulas Δ_0.

El relator de orden Para terminar nos ocupamos de un problema que, en principio, puede parecer meramente "antiestético", pero que, en realidad, cuando se profundiza en el estudio de la lógica de las teorías $\mathrm{I}\Sigma_n$ o AP, se convierte en un problema técnico. Se trata del hecho de que no es cierto que las fórmulas Δ_0 sólo contengan cuantificadores acotados, pues cada subfórmula $x \leq y$ contiene un cuantificador no acotado.

Vamos a demostrar que podemos añadir un relator diádico $\leq$ a $\mathcal{L}_a$ de modo que, con los axiomas adecuados, determine la relación de orden usual y de forma que la teoría así obtenida sea una extensión intrascendente de cualquier teoría aritmética dada (aunque por simplicidad trabajaremos con las teorías Σ_n).

Definición 4.32 Sea $\mathcal{L}_a^{\leq}$ el lenguaje formal que consta de los mismos signos que $\mathcal{L}_a$ más un relator diádico que representaremos por $\leq$.

Definimos las (semi)fórmulas $\Delta_0^{\leq}$ exactamente igual que en 4.9, salvo que las semifórmulas de tipo $t_1 \leq t_2$ no hay que entenderlas según la definición $x \leq u \equiv \bigvee u(x + u = y)$, sino que en ellas $\leq$ es el nuevo relator diádico.

Las fórmulas $\Sigma_n^{\leq}$ y $\Pi_n^{\leq}$ se definen exactamente igual, pero partiendo de las fórmulas $\Delta_0^{\leq}$.

Llamamos $\mathrm{I}\Sigma_n^{\leq}$ a las teorías cuyos axiomas son:

1. Los axiomas lógicos (que ahora incluyen a los que contienen el nuevo relator).
2. Los axiomas de AP distintos del principio de inducción.
3. El principio de inducción restringido a fórmulas de tipo $\Sigma_n^{\leq}$.
4. Los axiomas siguientes (en los que hay que entender que $\leq$ es el nuevo relator):

(a) $x \leq x$

(b) $x \leq y \wedge y \leq x \to x = y$

(c) $x \leq y \wedge y \leq z \to x \leq z$

(d) $x \leq y \vee y \leq x$

(e) $x \leq x + 1$

Vamos a probar que $\mathrm{I}\Sigma_n^{\leq}$ es una extensión intrascendente de $\mathrm{I}\Sigma_n$. Observemos que todas las fórmulas abiertas que no contienen el relator $\leq$ son $\Delta_0^{\leq}$, luego $\Sigma_1^{\leq}$, por lo que todos los teoremas de IA son también teoremas de $\mathrm{I}\Sigma_1^{\leq}$.

Veamos ahora que en $\mathrm{I}\Sigma_1^{\leq}$ podemos demostrar:

$$x \leq y \leftrightarrow \bigvee u(u + x = y).$$

Para probar la implicación $\to$ observamos que es equivalente a

$$\alpha(x) \equiv \bigvee u(x \leq y \to u + x = y),$$

que es una fórmula $\Sigma_1^{\leq}$, luego podemos razonar por inducción sobre x. Para $x = 0$ se cumple tomando $u = y$. Supuesto $\alpha(x)$, suponemos $x + 1 \leq y$, y los axiomas (e) y (c) nos dan entonces que $x \leq y$, luego, por hipótesis de inducción, existe un u tal que $u + x = y$. Si $u = 0$, entonces $x = y$, luego $x + 1 \leq x \leq x + 1$, luego, por (b), $x = x + 1$, luego $1 = 0$, en contra de AP1. Así pues, $u \neq 0$, y esto implica que existe un v tal que $u = v + 1$, luego $v + 1 + x = y$, que equivale a $v + (x + 1) = y$, y esto implica $\alpha(x + 1)$.

Para probar la implicación contraria suponemos que existe un u tal que $u + x = y$ y supongamos además que $y \leq x$. Entonces, por la implicación ya probada, existe un v tal que $v + y = x$, luego $u + v + y = y$, luego $u + v = 0$, luego $u = 0$, luego $x = y$, luego por (a) concluimos que $x \leq y$. Usando (d) resulta que, en cualquier caso, $x \leq y$.

Así pues, en $\mathrm{I}\Sigma_1^{\leq}$ se demuestra que $x \leq y$ es equivalente a la definición "antigua" de la relación de orden.

Ahora, para cada semifórmula α de $\mathcal{L}_a^{\leq}$ sin descriptores, definimos como sigue una semifórmula α^* de $\mathcal{L}_a$ (también sin descriptores):

1. $(t_1 = t_2)^* \equiv t_1 = t_2$,
2. $(t_1 \leq t_2)^* \equiv \bigvee u(u + t_1 = t_2)$,
3. $(\neg\alpha)^* \equiv \neg\alpha^*$,
4. $(\alpha \vee \beta)^* \equiv \alpha^* \vee \beta^*$,
5. $(\bigwedge u\alpha)^* \equiv \bigwedge u\, \alpha^*$,
6. $(\bigvee u\alpha)^* \equiv \bigvee u\, \alpha^*$.

En suma, α^* es la semifórmula que resulta de sustituir cada aparición del relator $\leq$ por la definición antigua de la relación de orden. En particular, si α no contiene el relator $\leq$, se cumple que $\alpha^* \equiv \alpha$. Notemos además que

$$(\alpha \to \beta)^* \equiv \alpha^* \to \beta^*, \quad (\alpha \wedge \beta)^* \equiv \alpha^* \wedge \beta^*, \quad (\alpha \leftrightarrow \beta)^* \equiv \alpha^* \leftrightarrow \beta^*.$$

A su vez, esto implica que si α es de tipo $\Sigma_n^{\leq}$, entonces α^* es de tipo Σ_n.

Una simple inducción[15] prueba que, para toda fórmula α de $\mathcal{L}_a^{\leq}$ sin descriptores, tenemos que

$$\underset{\mathrm{I}\Sigma_1^{\leq}}{\vdash} \alpha \leftrightarrow \alpha^*.$$

Si ϕ es un caso particular del principio de inducción para una fórmula α de tipo $\Sigma_n^{\leq}$, entonces ϕ^* es el principio de inducción para α^*, que es de tipo Σ_n, luego los axiomas de $\mathrm{I}\Sigma_n$ son teoremas de $\mathrm{I}\Sigma_n^{\leq}$, luego todos los teoremas de $\mathrm{I}\Sigma_n$ son teoremas de $\mathrm{I}\Sigma_n^{\leq}$.

Por otro lado, es fácil ver que si $\underset{\mathrm{I}\Sigma_n^{\leq}}{\vdash} \alpha$, entonces $\underset{\mathrm{I}\Sigma_n}{\vdash} \alpha^*$, pues, dada una demostración de α (que podemos tomar sin descriptores), sus axiomas lógicos se transforman en axiomas lógicos al aplicar $*$, sus axiomas propios se transforman en axiomas o en teoremas de $\mathrm{I}\Sigma_n$ y si una fórmula se sigue de fórmulas anteriores por una regla de inferencia, lo mismo sucede con sus transformaciones por $*$, luego el resultado es una demostración de α^*.

De aquí concluimos:

Teorema 4.33 *Toda fórmula de $\mathcal{L}_a^{\leq}$ es equivalente en $\mathrm{I}\Sigma_1$ a una fórmula de $\mathcal{L}_a$ y una fórmula de $\mathcal{L}_a$ es demostrable en $\mathrm{I}\Sigma_n^{\leq}$ (resp. $\mathrm{AP}^{\leq}$) si y sólo si lo es en $\mathrm{I}\Sigma_n$ (resp. AP).*

Demostración: Si α es una fórmula de $\mathcal{L}_a^{\leq}$, sabemos que es equivalente en IA a una fórmula $\bar{\alpha}$ sin descriptores, la cual es equivalente en $\mathrm{I}\Sigma_1$ a $\bar{\alpha}^*$, que es una fórmula de $\mathcal{L}_a$.

Por otra parte, si α es una fórmula cualquiera de $\mathcal{L}_a$, es equivalente en IA a una fórmula sin descriptores $\bar{\alpha}$. Si α es demostrable en $\mathrm{I}\Sigma_n^{\leq}$, también lo es $\bar{\alpha}$, luego $\bar{\alpha}^* \equiv \bar{\alpha}$ es demostrable en $\mathrm{I}\Sigma_n$, luego α también. ∎

Así pues, las teorías $\mathrm{I}\Sigma_n^{\leq}$ o $\mathrm{AP}^{\leq}$ tienen la misma capacidad expresiva o demostrativa que las teorías correspondientes $\mathrm{I}\Sigma_n$ o AP. Por ello, a partir de aquí no distinguiremos entre $\mathcal{L}_a$ y $\mathcal{L}_a^{\leq}$, ni entre $\mathrm{I}\Sigma_n$ e $\mathrm{I}\Sigma_n^{\leq}$, es decir, que es irrelevante si consideramos $\leq$ como definido con un particularizador o como un relator añadido, pero a efectos teóricos será útil contar con esta segunda posibilidad.

[15] Técnicamente, la formalización en ARP es como se describe en la nota al pie del teorema 3.25.

La jerarquía de Kleene reducida Hay otra variante técnica de las teorías $\mathrm{I}\Sigma_n$ que conviene tener en cuenta.

Definición 4.34 Las semifórmulas de $\mathcal{L}_a$ de tipo Δ_0^* son las que cumplen la definición 4.9 salvo que, en lugar de admitir cuantificadores acotados de la forma $\bigwedge u \le t\,\alpha$ y $\bigvee u \le t\,\alpha$, consideramos únicamente los de la forma $\bigwedge u \le x\,\alpha$ y $\bigvee u \le x\,\alpha$, donde x es una variable (libre o ligada) distinta de u.

Las fórmulas de tipo Σ_n^* o Π_n^* se definen como en 4.20, pero partiendo de fórmulas Δ_0^* en lugar de Δ_0.

Definimos $\mathrm{I}\Sigma_n^*$ como la teoría que resulta de restringir el principio de inducción de $\mathrm{I}\Sigma_n$ a fórmulas de tipo Σ_n^*. Ahora observamos que el principio de recolección 4.25 es válido con la misma prueba[16] en $\mathrm{I}\Sigma_1^*$ cuando la fórmula ϕ es de tipo Δ_0^*.

Usando esto, vamos a probar que toda fórmula de tipo Δ_0 es equivalente en $\mathrm{I}\Sigma_1^*$ a una fórmula de tipo Σ_1^* y a otra de tipo Π_1^*.

En efecto, para cada semifórmula α de tipo Δ_0 definimos como sigue dos semifórmulas α^+ y α^-, de tipo Σ_1^* y Π_1^*, respectivamente, con las mismas variables libres, de modo que (al sustituir las posibles variables ligadas que estén libres por variables libres) las tres fórmulas (la obtenida a partir de α, la de α^+ y la de α^-) son equivalentes en $\mathrm{I}\Sigma_1^*$.

1. $(t_1 = t_2)^+ \equiv (t_1 = t_2)^- \equiv t_1 = t_2$.
2. $(t_1 \le t_2)^+ \equiv (t_1 \le t_2)^- \equiv t_1 \le t_2$.
3. Si $\alpha^+ \equiv \bigvee u\,\delta^+$ y $\alpha^- \equiv \bigwedge u\,\delta^-$, donde δ^+ y δ^- son de tipo Δ_0^*, definimos

$$(\neg\alpha)^+ \equiv \bigvee u \neg\delta^-, \qquad (\neg\alpha)^- \equiv \bigwedge u\,\neg\delta^+.$$

   Claramente, si α (siempre cambiando las variables ligadas que estén libres) es equivalente a α^+ y a α^-, entonces $\neg\alpha$ es equivalente a $(\neg\alpha)^+$ y a $(\neg\alpha)^-$.

4. Si $\alpha^+ \equiv \bigvee u\,\delta^+$ y $\alpha^- \equiv \bigwedge u\,\delta^-$, $\beta^+ \equiv \bigvee u\,\epsilon^+$ y $\beta^- \equiv \bigwedge v\,\epsilon^-$, con δ^+, δ^-, ϵ^+, ϵ^- de tipo Δ_0^*, definimos

$$(\alpha \vee \beta)^+ \equiv \bigvee u(\delta^+ \vee \epsilon^+),$$

$$(\alpha \vee \beta)^- \equiv \bigwedge w \bigwedge uv \le w(w = \langle u, v\rangle_2 \to \delta^- \vee \epsilon^-).$$

   Así, si α es equivalente a α^+ y a α^- y β es equivalente a β^+ y a β^-, tenemos que $\alpha \vee \beta$ es equivalente a $(\alpha \vee \beta)^+$ y a $(\alpha \vee \beta)^-$.

[16] El único cambio es que en lugar de plantear una inducción sobre

$$\psi(z) \equiv z \le x + 1 \to \bigvee w \bigwedge u < z \bigvee v \le w\,\phi(u, v).$$

tomamos la fórmula Σ_1^* equivalente

$$\psi(z) \equiv \bigvee w(z \le x + 1 \to \bigwedge u \le z(u \ne z \to \bigvee v \le w\,\phi(u, v))).$$

5. Si $\alpha^+ \equiv \bigvee u\,\delta^+$ y $\alpha^- \equiv \bigwedge u\,\delta^-$, donde δ^+ y δ^- son de tipo Δ_0^*, observamos que si α es equivalente a α^+, entonces $\bigwedge v \le t\,\alpha$ es equivalente a cada una de las fórmulas siguientes:

   - $\bigwedge v \le t \bigvee u\,\delta^+$,
   - $\bigvee w(w = t \wedge \bigwedge v \le w \bigvee u\,\delta^+)$,
   - $\bigvee w(w = t \wedge \bigvee z \bigwedge v \le w \bigvee u \le z\,\delta^+)$, por recolección,
   - $\bigvee wz(w = t \wedge \bigwedge v \le w \bigvee u \le z\,\delta^+)$,
   - $\bigvee y \bigvee wz \le y(y = \langle w, z\rangle_2 \wedge w = t \wedge \bigwedge v \le w \bigvee u \le z\,\delta^+)$,

   y esta última fórmula es de tipo Σ_1^*, luego podemos tomarla como definición de $(\bigwedge v \le t\,\alpha)^+$.

   Similarmente, si α es equivalente a α^-, tenemos que $\bigwedge v \le t\,\alpha$ es equivalente a:

   - $\bigwedge v \le t \bigwedge u\,\delta^-$,
   - $\bigwedge w(w = t \to \bigwedge v \le w \bigwedge u\,\delta^-)$,
   - $\bigwedge wu(w = t \to \bigwedge v \le w\,\delta^-)$,
   - $\bigwedge y \bigwedge wu \le y(y = \langle w, u\rangle_2 \wedge w = t \to \bigwedge v \le w\,\delta^-)$,

   y la última fórmula es de tipo Π_1^*, luego podemos tomarla como definición de $(\bigwedge v \le t\,\alpha)^-$.

   Similarmente, $\bigvee v \le t\,\alpha$ es equivalente a

   - $\bigvee v \le t \bigvee u\,\delta^+$.
   - $\bigvee w(w = t \wedge \bigvee v \le w \bigvee u\,\delta^+)$,
   - $\bigvee wu(w = t \wedge \bigvee v \le w\,\delta^+)$,
   - $\bigvee y \bigvee wu \le y(y = \langle w, u\rangle_2 \wedge w = t \wedge \bigvee v \le w\,\delta^+)$,

   luego podemos tomar la última fórmula como definición de $(\bigvee v \le t\,\alpha)^+$. Por último, $\bigvee v \le t\,\alpha$ también equivale a

   - $\bigvee v \le t \bigwedge u\,\delta^-$,
   - $\bigwedge w(w = t \to \bigvee v \le w \bigwedge u\,\delta^-)$,
   - $\bigwedge w(w = t \to \neg \bigwedge v \le w \bigvee u\,\neg\delta^-)$,
   - $\bigwedge w(w = t \to \neg \bigvee z \bigwedge v \le w \bigvee u \le z\,\neg\delta^-)$ (por recolección),
   - $\bigwedge w(w = t \to \bigwedge z \bigvee v \le w \bigwedge u \le z\,\delta^-)$,
   - $\bigwedge wz(w = t \to \bigvee v \le w \bigwedge u \le z\,\delta^-)$,
   - $\bigwedge y \bigwedge wz \le y(y = \langle w, z\rangle_2 \wedge w = t \to \bigvee v \le w \bigwedge u \le z\,\delta^-)$,

   y tomamos la última fórmula como definición de $(\bigvee v \le t\,\alpha)^-$.

Una simple inducción (de la que ya hemos mostrado los casos no triviales para motivar la definición anterior) prueba que, en efecto, cada semifórmula α de tipo Δ_0 (siempre cambiando sus variables ligadas que estén libres) es equivalente en $\mathrm{I}\Sigma_1^*$ a α^+ y α^-. A su vez, si $\alpha^+ \equiv \bigvee u\,\delta^+$ y $\alpha^- \equiv \bigwedge u\,\delta^-$, donde δ^+ y δ^- son de tipo Δ_0^*, tenemos que

$$\bigvee v\,\alpha \leftrightarrow \bigvee vu\,\delta^+ \leftrightarrow \bigvee w \bigvee vu \le w(w = \langle v,u\rangle_2 \wedge \delta^+),$$

$$\bigwedge v\,\alpha \leftrightarrow \bigwedge vu\,\delta^- \leftrightarrow \bigwedge w \bigwedge vu \le w(w = \langle v,u\rangle_2 \to \delta^-).$$

Esto prueba que toda fórmula de tipo Σ_1 o Π_1 es equivalente en $\mathrm{I}\Sigma_1^*$ a una fórmula de tipo Σ_1^* o Π_1^*, respectivamente y a su vez esto implica inmediatamente el teorema siguiente:

Teorema 4.35 *Toda fórmula de tipo Σ_n o Π_n es equivalente en $\mathrm{I}\Sigma_1^*$ a una fórmula de tipo Σ_n^* o Π_n^*, respectivamente.*

Puesto que, trivialmente, toda fórmula de tipo Σ_n^* o Π_n^* es de tipo Σ_n o Π_n, respectivamente, podemos concluir que todos los axiomas de $\mathrm{I}\Sigma_n^*$ son axiomas de $\mathrm{I}\Sigma_n$ y que todos los axiomas de $\mathrm{I}\Sigma_n$ son teoremas de $\mathrm{I}\Sigma_n^*$.

Por consiguiente, si restringimos el principio de inducción de $\mathrm{I}\Sigma_n$ a fórmulas de tipo Σ_n^* obtenemos una teoría equivalente, en el sentido de que sus teoremas son los mismos. En lo sucesivo no distinguiremos entre $\mathrm{I}\Sigma_n$ o $\mathrm{I}\Sigma_n^*$.

4.4 Aritmética básica en $\mathrm{I}\Sigma_1$

Si comparamos ARP con $\mathrm{I}\Sigma_1$, vemos que en ARP tenemos "predefinidas" todas las funciones recursivas primitivas, pero sólo disponemos de la inducción sobre fórmulas abiertas (o a lo sumo Δ_0), mientras que en $\mathrm{I}\Sigma_1$ disponemos de la inducción sobre fórmulas Σ_1. En la sección siguiente veremos que esto último tiene más peso, en el sentido de que en $\mathrm{I}\Sigma_1$ es posible definir todas las funciones recursivas primitivas.

Más aún, veremos en [CS 1.24] que $\mathrm{I}\Sigma_1$ es también una extensión conservativa de ARP, de modo que, aun aceptando el principio de inducción sobre fórmulas Σ_1, tenemos la garantía de que cualquier resultado que demostremos expresable en $\mathcal{L}_{\mathrm{arp}}$ es demostrable en ARP. Esto ya no es cierto en el caso de $\mathrm{I}\Sigma_2$.

La mayor dificultad que tenemos que superar para definir las funciones recursivas primitivas es probar que en $\mathrm{I}\Sigma_1$ podemos identificar cada número natural con una sucesión finita, tal y como sabemos hacer en ARP, para lo cual tenemos que probar algunos resultados aritméticos.

Empezamos recordando que en 4.15 hemos introducido la relación de divisibilidad en AI, luego todos los resultados que hemos probado sobre ella valen también en $\mathrm{I}\Sigma_1$. Notemos además que

$$x \mid y \leftrightarrow \bigvee u \le y\, y = xu,$$

luego se trata de una fórmula Δ_0 (en el sentido amplio de que es equivalente a una fórmula Δ_0, o en sentido estricto si adoptamos ésta como definición).

Ahora podemos definir los números primos como en 2.33 (también con una fórmula Δ_0):

$$\text{primo}(p) \equiv p \neq 0 \wedge p \neq 1 \wedge \bigwedge uv \leq p(p = uv \to u = 1 \vee v = 1).$$

El teorema 2.34 puede probarse ahora esencialmente con el mismo argumento:

Teorema 4.36 $x > 1 \to \bigvee p \leq x(p \mid x \wedge \text{primo}(p))$.

DEMOSTRACIÓN: Si $x > 1$, como $x \mid x$, el teorema 4.29 nos da que existe un p tal que $p \neq 0 \wedge p \neq 1 \wedge p \mid x \wedge \bigwedge y < p\, y \nmid x$. En otras palabras, p es el menor divisor no trivial de x. En particular, $p \leq x$. Sólo tenemos que probar que es primo. Ciertamente, $p > 1$ y, si $p = uv$, pero $u \neq 1$ y $v \neq 1$, entonces $u \mid x$, $u \neq 0$, $u \neq 1$ y $u < p$, en contra de la minimalidad de p. ∎

Observemos que la definición de números *coprimos* es Δ_0:

$$\text{cop}(x, y) \equiv \bigwedge p \leq x(\text{primo}(p) \wedge p \mid x \to p \nmid y).$$

El teorema siguiente se prueba con la misma demostración que 2.35:

Teorema 4.37 $n \mid ab \wedge \text{cop}(n, a) \to n \mid b$.

Ahora usamos el principio de inducción fuerte 4.28 aplicado a la fórmula Δ_0:

$$\bigwedge abn \leq m(m = ab \wedge n \mid m \wedge \text{cop}(n, a) \to n \mid b)$$

Y esto a su vez nos permite probar 2.36 con el mismo argumento:

Teorema 4.38 $\text{primo}(p) \wedge p \mid ab \to p \mid a \vee p \mid b$.

Ahora podemos definir en $\mathrm{I}\Sigma_1$ una "relación de pertenencia" rudimentaria, suficiente para definir sucesiones finitas. En principio necesitaríamos definir el factorial de un número natural, pero, aunque no estamos en condiciones de hacerlo, el teorema siguiente nos da un "factorial rudimentario" que nos bastará:

Teorema 4.39 $\bigwedge x \bigvee y \bigwedge u(0 < u \leq x \to u \mid y)$.

DEMOSTRACIÓN: Por inducción[17] sobre x. Para $x = 0$ es trivial y si vale para x, tomamos un y divisible entre todo $0 < u \leq x$ y observamos que $y(x+1)$ cumple el teorema para $x + 1$. ∎

Abreviaremos[18] $u \in_0 (y, z) \equiv (1 + (1 + u)z) \mid y$.

Así, cada par de números naturales determina un conjunto, de tal forma que, como mostramos a continuación, todo conjunto finito definido por una fórmula está determinado por un par de números. En efecto:

[17] Notemos que la propiedad es Σ_1.

[18] Esta definición es Δ_0, pues equivale a $\bigvee st \leq y(y = st \wedge s = 1 + (1 + u)z)$.

Teorema 4.40 *Sea $\phi(u)$ una fórmula, que puede tener otras variables libres. Entonces la fórmula siguiente es un teorema de* AP *(o de $\mathrm{I}\Sigma_1$ si ϕ es Δ_1):*

$$\bigwedge x \bigvee yz \bigwedge u < x(u \in_0 (y,z) \leftrightarrow \phi(u))$$

Demostración: Si $x = 0$ el resultado es trivial, y si $x = 1$, basta tomar $y = z = 1$ si $\neg\phi(0)$ o bien $y = z = 0$ si $\phi(0)$. Por lo tanto, podemos suponer que $x > 1$. Sea z un número en las condiciones del teorema anterior, es decir, tal que $\bigwedge u(0 < u \le x \to u \mid z)$.

Veamos ahora que si $u < v < x$, entonces $\mathrm{cop}(1 + (1 + u)z, 1 + (1 + v)z)$. En efecto, supongamos que un primo p divide a ambos números. Abreviaremos $u_1 = 1 + u$, $v_1 = 1 + v$, de modo que $1 + u_1 z = ap$, $1 + v_1 z = bp$, para ciertos a, b. Notemos que $0 < v - u < x$, luego $v - u \mid z$. Por otra parte

$$1 + u_1 z \mid abp = a(1 + v_1 z), \qquad 1 + u_1 z \mid a(1 + u_1 z),$$

luego $1 + u_1 z$ también divide a la resta:

$$1 + u_1 z \mid a(v_1 - u_1)z = a(v - u)z.$$

Como claramente $\mathrm{cop}(1 + u_1 z, z)$, por 4.37 concluimos que

$$1 + u_1 z \mid a(v - u) \mid az.$$

Por el mismo argumento, $1 + u_1 z \mid a$, pero $1 + u_1 z = ap$, luego $1 + u_1 z = a$ y $p \mid 1$, contradicción.

Ahora probamos lo siguiente por inducción[19] sobre t:

$$\bigwedge t \le x \bigvee y(\bigwedge u < x((u < t \wedge \phi(u) \to u \in_0 (y,z))$$
$$\wedge (u \ge t \vee \neg\phi(u) \to \mathrm{cop}(y, 1 + (1 + u)z))).$$

Para $t = 0$ basta tomar $y = 1$. Supongamos el resultado cierto para t y sea y el número correspondiente. Podemos suponer que $t + 1 \le x$, o de lo contrario no hay nada que probar. Si $\neg\phi(t)$, entonces el mismo y cumple el resultado para $t + 1$. Supongamos, pues, $\phi(t)$ y llamemos $y' = y(1 + (1 + t)z)$. Entonces $t \in_0 (y', z)$, y si $u < t \wedge \phi(u)$ entonces $u \in_0 (y', z)$.

Tomamos por último un $u < x$ tal que $u \ge t + 1 \vee \neg\phi(u)$. Sabemos entonces que $\mathrm{cop}(y, 1 + (1 + u)z)$, y queremos probar lo mismo con y'. También sabemos que $\mathrm{cop}(y, 1 + (1 + t)z)$ y hemos probado que $1 + (1 + u)z$ y $1 + (1 + t)z$ son primos entre sí.

Entonces, si p es un primo tal que $p \mid y'$ y $p \mid 1 + (1 + u)t$, como $p \mid y'$, tiene que ser $p \mid y$ o bien $p \mid (1 + (1 + t)z)$, lo que contradice que $1 + (1 + u)z$ sea primo con estos dos números.

Con esto termina la inducción y, tomando $t = x$, resulta la fórmula del enunciado. ∎

[19] Notemos que la fórmula es Σ_1 si ϕ es Δ_1

Definición 4.41 Abreviaremos:

$$\mathrm{Suc}(y,z,x) \equiv \bigwedge u < x \bigvee v < y\, \langle u,v\rangle \in_0 (y,z).$$

Si se cumple esto, para cada $u < x$ llamaremos[20]

$$(y,z)_u \equiv \min v < y \mid \langle u,v\rangle \in_0 (y,z).$$

La fórmula $\mathrm{Suc}(y,z,x)$ significa que los números y, z codifican una sucesión de longitud x, la que a cada $u < x$ le asigna $(y,z)_u$. Así necesitamos dos números naturales para codificar una sucesión finita, lo cual es mucho menos eficiente que la codificación que hemos definido en ARP, pero con esta versión más tosca probaremos que todo lo que se puede hacer en ARP también se puede hacer en $\mathrm{I}\Sigma_1$, incluso codificar las sucesiones finitas elegantemente.

Terminamos esta sección probando que en $\mathrm{I}\Sigma_1$ podemos definir la exponencial 2^x. No vamos a necesitar esto, pues lo obtendremos indirectamente en la sección siguiente, donde veremos que en $\mathrm{I}\Sigma_1$ es posible definir todas las funciones recursivas primitivas, pero tiene interés ver un argumento directo que no dependa de ARP.

Definición 4.42 Abreviaremos[21]

$$\mathrm{SucExp}(y,z,x) \equiv x \geq 1 \wedge \mathrm{Suc}(y,z,x) \wedge (y,z)_0 = 1$$

$$\wedge \bigwedge u < x-1\ (y,z)_{u+1} = 2(y,z)_u.$$

$$\exp(x,v) \equiv \bigvee yz(\mathrm{SucExp}(y,z,x+1) \wedge (y,z)_x = v).$$

La fórmula $\mathrm{SucExp}(y,z,x)$ significa que y, z codifican una sucesión cuyo primer término es 1 y que cada uno se obtiene del anterior multiplicándolo por 2, de modo que se trata de la sucesión $1,2,4,8,\ldots,2^x$. Por lo tanto, $\exp(x,v)$ significa que $v = 2^x$.

Teorema 4.43 $\bigwedge x \overset{1}{\bigvee} v \exp(x,v)$.

Demostración: Si $\mathrm{SucExp}(y,z,x) \wedge \mathrm{SucExp}(y',z',x') \wedge x \leq x'$, entonces $\bigwedge u < x\ (y,z)_u = (y',z')_u$. Esto se prueba fácilmente por inducción[22] sobre u.

[20] Observemos que

$$\mathrm{Suc}(y,z,x) \leftrightarrow \bigwedge u < x \bigvee v w < y(w = \langle u,v\rangle \wedge w \in_0 (y,z)),$$

$$v = (y,z)_u \leftrightarrow \bigvee w \leq y(w = \langle u,v\rangle \wedge w \in_0 (y,z) \wedge \bigwedge v' < v(\neg \bigvee w \leq y \cdots)),$$

Vemos así que las dos fórmulas son Δ_0.

[21] Observemos que

$$\mathrm{SucExp}(y,z,x) \equiv x \geq 1 \wedge \mathrm{Suc}(y,z,x) \wedge \bigvee w \leq y(w = (y,z)_0 \wedge w = 1) \wedge$$

$$\bigwedge u < x-1 \bigvee w w' \leq y(w = (y,z)_{u+1} \wedge w' = (y,z)_u \wedge w = 2w'),$$

por lo que se trata de una fórmula Δ_0 y por lo tanto $\exp(x,v)$ es Σ_1.

[22] la fórmula es Δ_0

Si $u = 0$, entonces $(y,z)_0 = 1 = (y',z')_0$. Si es cierto para u y $u+1 < x$, entonces

$$(y,z)_{u+1} = 2(y,z)_u = 2(y',z')_u = (y',z')_{u+1}.$$

Esto nos da la unicidad. Veamos ahora que si $\mathrm{SucExp}(y,z,x)$ entonces $u < x$ implica $u < (y,z)_u$. Por inducción[23] sobre u. Para $u = 0$ es $0 < 1 = (y,z)_0$. Si es cierto para u, es decir, si $u < (y,z)_u$ y $u+1 < x$, entonces

$$u + 1 \leq 1 \cdot (y,z)_u < 2(y,z)_u = (y,z)_{u+1}.$$

Similarmente, si $u < v < x$, entonces $(y,z)_u < (y,z)_v$. Por inducción[24] sobre v. Si $v = 0$ no hay nada que probar. Si vale para v y se cumple $u < v+1$, entonces $u \leq v$. Si $u = v$ tenemos que

$$(y,z)_u < 2(y,z)_u = (y,z)_{v+1}.$$

Si $u < v$ tenemos $(y,z)_u < (y,z)_v < 2(y,z)_v = (y,z)_{v+1}$.

Pasemos ya a probar la existencia. Para ello probamos por inducción[25] sobre x que $\bigwedge x \geq 1 \bigvee yz\ \mathrm{SucExp}(y,z,x)$.

En efecto, para $x = 0$ es trivial. Si vale para x, tratamos aparte el caso $x = 0$ (es decir, probamos aparte que el resultado vale para $x = 1$). Para ello aplicamos 4.40, que nos da:

$$\bigvee yz \bigwedge u < 2(u \in_0 (y,z) \leftrightarrow u = 1).$$

Teniendo en cuenta que $\langle 0,0\rangle = 0$ y $\langle 0,1\rangle = 1$, esto significa que si tomamos y, z que cumplan esto, se cumple $\mathrm{Suc}(y,z,1)$ y $(y,z)_0 = 1$, luego $\mathrm{SucExp}(y,z,1)$.

Supongamos ahora que $x \geq 1$ y que existen y, z tales que $\mathrm{SucExp}(y,z,x)$. Sea $q = (y,z)_{x-1}$ y sea $q' = \langle x, 2q\rangle$. Por 4.40 existen y', z' tales que

$$\bigwedge u \leq q'(u \in_0 (y',z') \leftrightarrow (u \in_0 (y,z) \wedge \bigvee i < x \bigvee v \leq u\ u = \langle i,v\rangle) \vee u = \langle x, 2q\rangle).$$

Se cumple que $\mathrm{Suc}(y',z',x+1)$. En efecto, si $i < x+1$, o bien $i < x$, en cuyo caso, por la monotonía que hemos probado,

$$u = \langle i, (y,z)_i\rangle \leq \langle x, q\rangle \leq q',$$

luego $u \in_0 (y',z')$, o bien $i = x$, en cuyo caso $u = \langle x, 2q\rangle = q'$ también cumple $u \in_0 (y',z')$, luego en cualquier caso $\bigwedge i < x+1 \bigvee v\, \langle i,v\rangle \in_0 (y',z')$. Más aún, si $\langle i,v\rangle \in_0 (y',z')$ con $i < x$, tiene que ser $v \geq (y,z)_i$, pues en caso contrario $v < (y,z)_i$, pero entonces $\langle i,v\rangle \leq \langle i,(y,z)_i\rangle \leq q'$, luego $\langle i,v\rangle \in_0 (y,z)$, pero entonces $(y,z)_i \leq v$, contradicción. En otros términos,

$$\bigwedge i < x\ (y',z')_i = (y,z)_i.$$

Similarmente se ve que $(y',z')_x = 2q = 2(y,z)_{x-1}$, y de aquí se sigue inmediatamente que $\mathrm{SucExp}(y',z',x+1)$. Esto completa la inducción y, para cada x, tenemos que existen y, z tales que $\mathrm{SucExp}(y,z,x+1)$, luego $\bigvee v \exp(x,v)$. ∎

[23] La fórmula es $u < x \to \bigvee v < y(u < v \wedge v = (y.z)_u)$, luego es Δ_0.
[24] La fórmula es $\bigvee ww' \leq y(w = (y,z)_u \wedge w' = (y,z)_v \wedge w < w')$, luego es Δ_0.
[25] La fórmula es Σ_1.

Definición 4.44 Definimos[26] $2^x \equiv v \mid \exp(x,v)$.

Claramente se cumple:

$$2^0 = 1 \wedge \bigwedge x(2^{x+1} = 2^x \cdot 2).$$

Más aún, en la prueba del teorema anterior hemos visto también las dos primeras de las tres propiedades siguientes (y la tercera se prueba sin problemas por inducción[27] sobre y):

$$\bigwedge x\ x < 2^x, \qquad \bigwedge uv(u < v \to 2^u < 2^v), \qquad \bigwedge xy\ 2^{x+y} = 2^x \cdot 2^y.$$

El teorema siguiente demuestra esencialmente que cada número determina sus cifras binarias:

Teorema 4.45 $\bigwedge xy\overset{1}{\bigvee} uvw(v \leq 1 \wedge w < 2^x \wedge y = 2^{x+1} \cdot u + 2^x \cdot v + w)$.

Demostración: Por la división euclídea, existen u y $q < 2^{x+1}$ tales que $y = 2^{x+1} \cdot u + q$. A su vez, existen v y $w < 2^x$ tales que $q = 2^x v + w$, pero tiene que ser $v < 2$, pues en caso contrario $q \geq 2^{x+1}$. Esto nos da la existencia. Si tenemos

$$2^{x+1} \cdot u + 2^x \cdot v + w = 2^{x+1} \cdot u' + 2^x \cdot v' + w',$$

con $v, v' \leq 1$, entonces $w = w'$, pues ambos son el resto de la división euclídea entre 2^x, luego

$$2^{x+1} \cdot u + 2^x \cdot v = 2^{x+1} \cdot u' + 2^x \cdot v'$$

y de aquí $2u + v = 2u' + v'$, pero entonces $v = v'$, pues ambos son el resto de la división euclídea entre 2, luego $u = u'$. ∎

El número v dado por el teorema anterior es la cifra que ocupa la posición x en la expresión binaria de y o, simplemente, el bit x-ésimo de y:

Definición 4.46 Abreviaremos:[28]

$$\mathrm{bit}(x,y) \equiv v \mid \bigvee uw(v \leq 1 \wedge w < 2^x \wedge y = 2^{x+1} \cdot u + 2^x \cdot v + w).$$

Nota: La relación de pertenencia A partir de aquí podemos definir

$$x \in y \equiv \mathrm{bit}(x,y) = 1, \qquad x \notin y \equiv \mathrm{bit}(x,y) = 0,$$

que es la misma relación de pertenencia que ya habíamos definido en ARP, y es un ejercicio de pura aritmética demostrar en $\mathrm{I}\Sigma_1$ los resultados sobre ella que ya hemos probado en ARP. No vamos a hacerlo aquí, pues en la sección siguiente lo obtendremos de forma indirecta. ∎

[26] Tenemos que el término 2^x es Σ_1, porque $v = 2^x$ equivale a $\exp(x,v)$, que es claramente Σ_1, luego 2^x es Δ_1 por la observación tras la definición 4.22.

[27] La fórmula es Δ_1, pues resulta de introducir los términos 2^{x+y}, 2^x, 2^y (de tipo Δ_1) en la fórmula $u = vw$ (trivialmente Δ_1).

[28] Se trata de un término Δ_1, pues $v = \mathrm{bit}(x,y)$ resulta de introducir los términos 2^{x+1} y 2^x (de tipo Δ_1) en una fórmula Σ_1.

4.5 Funciones recursivas primitivas

Vamos a ver que en $\mathrm{I}\Sigma_1$ es posible definir todas las funciones recursivas primitivas. Nos apoyaremos en el teorema siguiente:

Teorema 4.47 *Sean $\phi(x_1,\dots,x_n,y)$ y $\psi(x_1,\dots,x_n,x,s,y)$ dos fórmulas de $\mathcal{L}_a$ de tipo Σ_1 con a lo sumo las variables libres indicadas, tales que*

$$\underset{\mathrm{I}\Sigma_1}{\vdash} \overset{1}{\bigvee} y\,\phi(x_1,\dots,x_n,y), \qquad \underset{\mathrm{I}\Sigma_1}{\vdash} \overset{1}{\bigvee} y\,\psi(x_1,\dots,x_n,x,y',y).$$

Entonces existe otra fórmula $\chi(x_1,\dots,x_n,x,y)$ de $\mathcal{L}_a$ de tipo Σ_1, con a lo sumo las variables libres indicadas, tal que

$$\underset{\mathrm{I}\Sigma_1}{\vdash} \overset{1}{\bigvee} y\,\chi(x_1,\dots,x_n,x,y), \qquad \underset{\mathrm{I}\Sigma_1}{\vdash} \bigvee y(\chi(x_1,\dots,x_n,0,y) \wedge \phi(x_1,\dots,x_n,y)),$$

$$\underset{\mathrm{I}\Sigma_1}{\vdash} \bigvee yy'(\chi(x_1,\dots,x_n,x+1,y) \wedge \chi(x_1,\dots,x_n,x,y') \wedge \psi(x_1,\dots,x_n,x,y',y)).$$

Si pensamos que las fórmulas ϕ y ψ definen dos funciones, entonces χ define la función definida por recursión a partir de dichas funciones.

Demostración: Por brevedad omitimos las variables $x_1,\dots,x_n$. Basta definir

$$\chi(x,y) \equiv \bigvee uv(\mathrm{Suc}(u,v,x+1) \wedge \phi((u,v)_0) \wedge$$
$$\bigwedge i < x\,\psi(i,(u,v)_i,(u,v)_{i+1}) \wedge y = (u,v)_x).$$

Claramente se trata de una fórmula de tipo Σ_1. Vamos a probar que cumple lo requerido. En primer lugar probamos por inducción sobre x que $\bigvee y\,\chi(x,y)$ (la fórmula es obviamente Σ_1). Consideramos el único y_0 que cumple $\phi(y_0)$ y sea $p = \langle 0, y_0\rangle$. Por el teorema 4.40 existen u, v tales que

$$\bigwedge i < p+1(i \in_0 (u,v) \leftrightarrow i = p).$$

Claramente $\mathrm{Suc}(u,v,1)$ y $(u,v)_0 = y_0$, luego $\chi(0,y_0)$, luego $\bigvee y\,\chi(0,y)$.

Supongamos ahora que existe un y' tal que $\chi(x,y')$. Sean u', v' según la definición. Sea y el único número natural que cumple $\psi(x,y',y)$. Por el principio de recolección existe un w tal que

$$\bigwedge i < x+1\ \langle i,(u',v')_i\rangle < w,$$

y claramente podemos exigir además que $\langle x+1, y\rangle < w$.

Por el teorema 4.40 existen u, v tales que

$$\bigwedge p < w(p \in_0 \langle u,v\rangle \leftrightarrow (\bigvee i < x+1\ p = \langle i,(u',v')_i\rangle) \vee p = \langle x+1,y\rangle).$$

Es claro que $\mathrm{Suc}(u,v,x+2))$, así como que

$$\bigwedge i < x+1\,(u,v)_i = (u',v')_i, \qquad (u,v)_{x+1} = y.$$

Es fácil ver entonces que $\chi(x+1,y)$.

Ahora veamos que $\bigvee^1 y\,\chi(x,y)$. Como ya hemos probado la existencia, basta ver que si $\chi(x,y)$ y $\chi(x,\bar{y})$, entonces $y=\bar{y}$. Sean $u,v,\bar{u},\bar{v}$ según la definición de χ y veamos por inducción sobre i que $i<x+1\to(u,v)_i=(\bar{u},\bar{v})_i$. En efecto, como $\phi((u,v)_0)$ y $\phi(\bar{u},\bar{v})_0$, la unicidad de ϕ implica que $(u,v)_0=(\bar{u},\bar{v})_0$.

Supuesto cierto que $(u,v)_i=(\bar{u},\bar{v})_i$, tenemos que

$$\psi(i,(u,v)_i,(u,v)_{i+1})\wedge\psi(i,(\bar{u},\bar{v})_i,(\bar{u},\bar{v})_{i+1})$$

y la unicidad de ψ implica que $(u,v)_{i+1}=(\bar{u},\bar{v})_{i+1}$. En particular, tenemos que $y=(u,v)_x=(\bar{u},\bar{v})_x=\bar{y}$.

Ahora veamos que χ cumple lo que dice el enunciado. Antes hemos visto que el único y_0 que cumple $\phi(y_0)$ cumple también $\chi(0,y_0)$, luego $\bigvee y(\chi(0,y)\wedge\phi(y))$, que es la primera condición que teníamos que probar.

Tomemos ahora el único y que cumple $\chi(x+1,y)$ y sean u,v según la definición de χ. Es claro que estos valores cumplen también la definición de $\chi(x,y)$, de donde se sigue que $y'=(u,v)_x$ cumple $\chi(x,y')$. Pero de $\chi(x+1,y)$ se sigue $\psi(x,(u,v)_x,(u,v)_{x+1})$, es decir, $\psi(x,y',y)$. Con esto hemos probado que

$$\bigvee yy'(\chi(x+1,y)\wedge\chi(x,y')\wedge\psi(x,y',y)).$$

■

Nota Por el teorema 3.29, si las fórmulas ϕ y ψ no tienen descriptores, podemos exigir que χ tampoco los tenga y que las demostraciones en $\mathrm{I}\Sigma_1$ que proporciona el teorema anterior no contengan descriptores. ■

En 4.20 hemos señalado que la jerarquía de Kleene puede definirse para fórmlas de $\mathcal{L}_{\mathrm{arp}}$ tomando como fórmulas Δ_0 las fórmulas atómicas, y que esta jerarquía cumple igualmente el teorema 4.21. Así podemos hablar de fórmulas Σ_n y Π_n de $\mathcal{L}_{\mathrm{arp}}$ en sentido estricto (las que satisfacen la definición) y de fórmulas en sentido amplio, que son las equivalentes en ARP a fórmulas del tipo correspondiente en sentido amplio. A las fórmulas que son a la vez Σ_n y Π_n en sentido amplio las llamamos fórmulas Δ_n.

Definición 4.48 Llamaremos $\mathrm{I}\Sigma_n^+$ (resp. AP^+) a la teoría axiomática sobre $\mathcal{L}_{\mathrm{arp}}$ (que no tiene descriptor) cuyos axiomas son los de ARP con el principio de inducción extendido a fórmulas de tipo $\mathrm{I}\Sigma_n$ en $\mathcal{L}_{\mathrm{arp}}$ (resp. a fórmulas arbitrarias).

En principio tomamos como axiomas de inducción los determinados por fórmulas del tipo correspondiente en sentido estricto, pero es claro que cualquier fórmula de inducción asociada a una fórmula de tipo Σ_n en sentido amplio es equivalente a la correspondiente a cualquier fórmula Σ_n en sentido estricto equivalente, luego es un teorema de $\mathrm{I}\Sigma_n^+$.

Así $\mathrm{I}\Sigma_n^+$ (resp. AP^+) es obviamente una extensión de ARP, pero también lo es de $\mathrm{I}\Sigma_n$ (resp. AP) (considerando a estas teorías sin descriptor). En efecto, todos los axiomas de AP menos el de inducción son axiomas de ARP, y el

principio de inducción para fórmulas de tipo Σ_n en $\mathcal{L}_{\rm arp}$ extiende al principio de inducción de $\mathrm{I}\Sigma_n$, que sólo admite fórmulas de $\mathcal{L}_a$. Por lo tanto, todo teorema sin descriptores de $\mathrm{I}\Sigma_n$ (resp. AP) es un teorema de $\mathrm{I}\Sigma_n^+$ (resp. AP^+).

El objetivo de esta sección es probar el teorema siguiente:

Teorema 4.49 *Toda fórmula de $\mathcal{L}_{\rm arp}$ es equivalente en $\mathrm{I}\Sigma_1^+$ a una fórmula de $\mathcal{L}_a$, que será de tipo Σ_n, Π_n o Δ_n (en $\mathrm{I}\Sigma_1$) si lo es la fórmula dada. Si una fórmula de $\mathcal{L}_a$ (sin descriptores) es demostrable en $\mathrm{I}\Sigma_n^+$, también es demostrable en $\mathrm{I}\Sigma_n$. Así, $\mathrm{I}\Sigma_n^+$ (resp. AP^+) es una extensión intrascendente de $\mathrm{I}\Sigma_n$ (resp. AP).*

Esto significa que los funtores de $\mathcal{L}_{\rm arp}$ y sus definiciones no aportan nada a $\mathrm{I}\Sigma_n$, pues pueden definirse en $\mathrm{I}\Sigma_1$ de forma que todo lo que se puede probar con los funtores adicionales se puede probar también sin ellos.

La prueba de este teorema es muy similar a la del teorema 3.29 de eliminación de descriptores, así que detallaremos únicamente los pasos en los que hay diferencias sustanciales. Como aquí consideramos lenguajes sin descriptores, el argumento se simplifica un poco.

Empezamos asociando a cada funtor n-ádico f de $\mathcal{L}_{\rm arp}$ una fórmula sin descriptores $\phi_f(x_1,\dots,x_n,y)$ de tipo Σ_1 en $\mathcal{L}_a$ cuyas variables libres están entre las indicadas. Lo hacemos por recurrencia sobre f.

1. Si $f \equiv S$, entonces $\phi_S(x_1,y) = y = Sx_1$.
2. Si $f \equiv c$, entonces $\phi_x(x_1,y) = y = 0$.
3. Si $f \equiv p_i^n$, entonces $\phi_{p_i^n}(x_1,\dots,x_n,y) \equiv y = x_i$.
4. Si $f \equiv \kappa(h,g_1,\dots,g_m)$, entonces

$$\phi_f(x_1,\dots,x_n,y) \equiv \bigvee u_1\cdots u_m(\phi_{g_1}(x_1,\dots,x_n,u_1) \wedge \cdots \wedge \phi_{g_m}(x_1,\dots,x_n,u_m)$$
$$\wedge\, \phi_h(u_1,\dots,u_m,y)).$$

5. Si $f \equiv \rho(g,h)$, tomamos como ϕ_f la fórmula dada por el teorema 4.47 a partir de ϕ_g y ϕ_h. De este modo:

$$\vdash_{\mathrm{I}\Sigma_1} \overset{1}{\bigvee} y\, \phi_f(x_1,\dots,x_n,x,y), \qquad \vdash_{\mathrm{I}\Sigma_1} \bigvee y(\phi_f(x_1,\dots,x_n,0,y) \wedge \phi_g(x_1,\dots,x_n,y)),$$

$$\vdash_{\mathrm{I}\Sigma_1} \bigvee yy'(\phi_f(x_1,\dots,x_n,x{+}1,y) \wedge \phi_f(x_1,\dots,x_n,x,y') \wedge \phi_h(x_1,\dots,x_n,x,y',y)).$$

El último punto es correcto porque es fácil razonar inductivamente que, para todo funtor f, se cumple

$$\vdash_{\mathrm{I}\Sigma_1} \overset{1}{\bigvee} y\, \phi_f(x_1,\dots,x_n,y).$$

De este modo, si ϕ_g y ϕ_h cumplen este resultado de unicidad, podemos aplicarles el teorema 4.47 para obtener una fórmula ϕ_f que también cumple la unicidad requerida. De aquí se sigue además que ϕ_f es de tipo Δ_1 en $\mathrm{I}\Sigma_1$.

Ahora asociamos a cada semitérmino t de $\mathcal{L}_{\mathrm{arp}}$ una semifórmula de $\mathcal{L}_a$ sin descriptores $\psi_t(y)$ de tipo Σ_1 con las mismas variables libres más una variable adicional y.

1. Si $t \equiv x$ es una variable (libre o ligada), entonces $\psi_t(y) \equiv y = x$.
2. Si $t \equiv 0$, entonces $\psi_t(y) \equiv y = 0$.
3. Si $t \equiv f(t_1, \ldots, t_n)$, definimos
$$\psi_t(y) \equiv \bigvee u_1 \cdots u_n(\psi_{t_1}(u_1) \wedge \cdots \wedge \psi_{t_n}(u_n) \wedge \phi_f(u_1, \ldots, u_n, y)).$$

Finalmente definimos, para cada semifórmula α de $\mathcal{L}_{\mathrm{arp}}$, una semifórmula α^* de $\mathcal{L}_a$ con a lo sumo las mismas variables libres:

1. Si $\alpha \equiv (t_1 = t_2)$, entonces $\alpha^* \equiv \bigvee u(\psi_{t_1}(u) \wedge \psi_{t_2}(u))$.
2. Si $\alpha \equiv \neg\beta$, entonces $\alpha^* \equiv \neg\beta^*$.
3. Si $\alpha \equiv \beta \vee \gamma$, entonces $\alpha^* \equiv \beta^* \vee \gamma^*$, y esto implica a su vez que $(\beta \to \gamma)^* \equiv \beta^* \to \gamma^*$, $(\beta \wedge \gamma)^* \equiv \beta^* \wedge \gamma^*$, $(\beta \leftrightarrow \gamma)^* \equiv (\beta^* \leftrightarrow \gamma^*)$.
4. Si $\alpha \equiv \bigwedge u\, \beta$, entonces $\alpha^* \equiv \bigwedge u\, \beta^*$.
5. Si $\alpha \equiv \bigvee u\, \beta$, entonces $\alpha^* \equiv \bigvee u\, \beta^*$.

Vamos a ir probando varios hechos sobre los conceptos que acabamos de introducir.

1. Para todo funtor f,
$$\underset{\mathrm{I}\Sigma_1^+}{\vdash} y = f(x_1, \ldots, x_n) \leftrightarrow \phi_f(x_1, \ldots, x_n, y).$$
Si $f \equiv S, +, \cdot$, la equivalencia se demuestra, de hecho, en $\mathrm{I}\Sigma_1$.
La equivalencia en $\mathrm{I}\Sigma_1^+$ se demuestra por inducción sobre la longitud de f. Todos los casos son obvios salvo el correspondiente a $f \equiv \rho(g, h)$. En este caso hay que probar:
$$\bigwedge y(y = f(x_1, \ldots, x_n, x) \leftrightarrow \phi_f(x_1, \ldots, x_n, x, y)).$$
Razonamos por inducción sobre x. Notemos que la fórmula es de tipo Π_1 en $\mathcal{L}_{\mathrm{arp}}$, por lo que la inducción es lícita en $\mathrm{I}\Sigma_1^+$.
Para $x = 0$ tenemos, por la definición de f, que $y = f(x_1, \ldots, x_n, 0)$ equivale a $y = g(x_1, \ldots, x_n)$. Por otro lado, por la construcción de ϕ_f,
$$\bigvee y(\phi_f(x_1, \ldots, x_n, 0, y) \wedge \phi_g(x_1, \ldots, x_n, y)),$$
y por hipótesis de inducción la segunda fórmula equivale a $y = g(x_1, \ldots, x_n)$. Esto implica que $g(x_1, \ldots, x_n)$ es el único y que cumple $\phi_f(x_1, \ldots, x_n, 0, y)$, luego $\phi_f(x_1, \ldots, x_n, 0, y)$ equivale a $y = g(x_1, \ldots, x_n)$.

Supuesto que la equivalencia es cierta para x, por la definición de f tenemos (omitiendo, por brevedad, $x_1, \ldots, x_n$) que $y = f(Sx)$ equivale a

$$y = h(x, f(x)).$$

Por otro lado, por construcción de ϕ_f tenemos que

$$\bigvee yy'(\phi_f(Sx, y) \wedge \phi_f(x, y') \wedge \phi_h(x, y', y)).$$

La hipótesis de inducción sobre la longitud de f nos da que $\phi_h(x, y', y)$ equivale a $y = h(x, y')$, y la hipótesis de inducción sobre x nos da que $\phi_f(x, y')$ equivale a $y' = f(x)$, luego $\phi_f(Sx, y)$ equivale a $y = h(x, f(x))$.

En el caso de $f \equiv S$ es inmediato que la equivalencia puede probarse en $\mathrm{I}\Sigma_1$. Consideremos ahora el caso en que $f \equiv + \equiv \rho(p_1^1, \kappa(S, p_2^2))$. Llamemos $f \equiv \kappa(S, p_2^2)$, de modo que

$$\phi_f(x_1, x_2, y) \equiv \bigvee u(\phi_{p_2^2}(x_1, x_2, u) \wedge \phi_S(y, u)) \equiv \bigvee u(u = x_2 \wedge y = Su),$$

y esto equivale en $\mathrm{I}\Sigma_1$ a que $y = Sx_2$.

Ahora, para $\phi_+(x_1, x_2, y)$ tenemos

$$\bigvee y(\phi_+(x_1, 0, y) \wedge \phi_{p_1^1}(x_1, y)) \equiv \bigvee y(\phi_+(x_1, 0, y) \wedge y = x_1),$$

$$\bigvee yy'(\phi_+(x_1, Sx_2, y) \wedge \phi_+(x_1, x_2, y') \wedge \phi_f(x_1, y', y)),$$

y estas fórmulas equivalen, respectivamente, a $\phi_+(x_1, 0, x_1)$ y a

$$\bigvee y'(\phi_+(x_1, Sx_2, Sy') \wedge \phi_+(x_1, x_2, y')).$$

De aquí se sigue por inducción sobre x_2 que

$$\bigwedge y(y = \phi_+(x_1, x_2, y) \leftrightarrow y = x_1 + x_2).$$

En efecto, para $x_2 = 0$ tenemos $\phi_+(x_1, 0, x_1)$ y, por la unicidad de ϕ_+, esto implica que $\phi_+(x_1, 0, y) \leftrightarrow y = x_1 = x_1 + 0$. Si vale para x_0, en la segunda fórmula que hemos obtenido tenemos que $y' = x_1 + x_2$, luego nos da que $\phi_+(x_1, Sx_2, S(x_1 + x_2))$, y la unicidad implica que

$$\phi_+(x_1, Sx_2, y) \leftrightarrow y = x_1 + Sx_2.$$

El caso del producto se razona análogamente.

2. Si $t \equiv f(x_1, \ldots, x_n)$, entonces $\underset{\mathrm{I}\Sigma_1}{\vdash} \psi_t(y) \leftrightarrow \phi_f(y)$.

   La comprobación es trivial, así como la del hecho siguiente:

3. $\underset{\mathrm{I}\Sigma_1}{\vdash} \overset{1}{\bigvee} y\, \psi_t(y)$, $\qquad \underset{\mathrm{I}\Sigma_1^+}{\vdash} (y = t \leftrightarrow \psi_t(y))$.

   Si t está en $\mathcal{L}_a$, la segunda equivalencia puede probarse en $\mathrm{I}\Sigma_1$.

Esto se debe a que el caso 3 de la definición de ψ_t sólo puede darse con $f \equiv S$, $f \equiv +$ o $f \equiv \cdot$ y, considerando por ejemplo el segundo caso,

$$\psi_{t_1+t_2}(y) \equiv \bigvee u_1 u_2(\psi_{t_1}(u_1) \wedge \psi_{t_2}(u_2) \wedge y = u_1 + u_2).$$

Si suponemos, como hipótesis inductiva, que $\psi_{t_i}(y)$ equivale en $\mathrm{I}\Sigma_1$ a $y = t_i$, esto equivale a que $y = t_1 + t_2$. El caso del producto es análogo y el de S es inmediato.

4. Como las fórmulas ψ_t son Σ_1, por la unicidad son de hecho de tipo Δ_1 en $\mathrm{I}\Sigma_1$, ya que
$$\psi_t(x) \leftrightarrow \bigwedge u(\psi_t(u) \to x),$$
y esto a su vez implica que si α es Δ_0 en $\mathcal{L}_{\mathrm{arp}}$ (es decir, atómica), entonces α^* es de tipo Δ_1 en $\mathrm{I}\Sigma_1$, pues α^* es claramente de tipo Σ_1 y
$$\alpha^* \leftrightarrow \bigwedge u(\psi_{t_1}(u) \wedge \psi_{t_2}(u)).$$
De aquí se sigue a su vez que si α es de tipo Σ_n, Π_n o Δ_n en $\mathcal{L}_{\mathrm{arp}}$ (para cierto $n \geq 1$) entonces α^* es del tipo correspondiente en $\mathrm{I}\Sigma_1$ (pero hay que tener presente que las fórmulas Δ_0 en $\mathcal{L}_{\mathrm{arp}}$ se corresponden con fórmulas Δ_1 en $\mathrm{I}\Sigma_1$).

5. Para toda fórmula α de $\mathcal{L}_{\mathrm{arp}}$, se cumple
$$\underset{\mathrm{I}\Sigma_1^+}{\vdash} (\alpha \leftrightarrow \alpha^*).$$
Más aún si α está en $\mathcal{L}_a$, la equivalencia se puede probar en $\mathrm{I}\Sigma_1$.

6. Si t, t' son términos de $\mathcal{L}_{\mathrm{arp}}$, α es una fórmula y x es una variable que no esté en $\mathtt{S}_y^t t'$ ni en $\mathtt{S}_y^t \alpha$, entonces
$$\underset{\mathrm{I}\Sigma_1}{\vdash} \psi_{\mathtt{S}_y^t t'}(y) \leftrightarrow \bigvee u(\psi_t(u) \wedge \mathtt{S}_y^u \psi_{t'}(y)), \quad \underset{\mathrm{I}\Sigma_1}{\vdash} (S_y^t \alpha)^* \leftrightarrow \bigvee u(\psi_t(u) \wedge \mathtt{S}_y^u \alpha^*).$$
La prueba es idéntica a la del hecho análogo en la prueba del teorema 3.29.

7. Si α es un axioma de $K_{\mathcal{L}_{\mathrm{arp}}}$, entonces $\underset{\mathrm{I}\Sigma_1}{\vdash} \alpha^*$.
La prueba es también idéntica a la vista en la prueba del teorema 3.29 (salvo que no necesitamos considerar los axiomas asociados al descriptor.)

8. Si α es la definición de un funtor de ARP, entonces $\underset{\mathrm{I}\Sigma_1}{\vdash} \alpha^*$.
En efecto, si $\alpha \equiv c(x) = 0$, entonces
$$\alpha^* \equiv \bigvee u(\psi_{c(x)}(u) \wedge \phi_0(u)),$$
donde $\psi_{c(x)}(u)$ equivale a $\phi_c(u) \equiv u = 0$, con lo que α^* equivale a
$$\bigvee u(u = 0 \wedge u = 0),$$
que ciertamente es un teorema de $\mathrm{I}\Sigma_1$.

Si $\alpha \equiv p_i^n(x_1,\dots,x_n) = x_i$, entonces α^* equivale a

$$\bigvee u(\phi_{p_i^n}(u) \wedge \psi_{x_i}(u)),$$

que a su vez equivale a $\bigvee u(u = x_i \wedge u = x_i)$, que es trivialmente un teorema.

Si $\alpha \equiv f(x_1,\dots,x_n) = h(g_1(x_1,\dots,x_n),\dots,g_m(x_1,\dots,x_n))$, entonces α^* equivale a

$$\bigvee u(\phi_f(u) \wedge \bigvee u_1\cdots u_n(\phi_{g_1}(u_1) \wedge \cdots \wedge \phi_{g_m}(u_m) \wedge \phi_h(u_1,\dots,u_n,u))),$$

que a su vez equivale a $\bigvee u(\phi_f(u) \wedge \phi_f(u))$, lo cual es un teorema de $\mathrm{I}\Sigma_1$.

Si $\alpha \equiv f(x_1,\dots,x_n,0) = g(x_1,\dots,x_n)$, entonces α^* equivale a

$$\bigvee u(\phi_f(x_1,\dots,x_n,0,u) \wedge \phi_g(x_1,\dots,x_n,u)),$$

y esto es un teorema de $\mathrm{I}\Sigma_1$ por la construcción de ϕ_f. Finalmente, si

$$\alpha \equiv f(x_1,\dots,x_n,Sx) = h(x_1,\dots,x_n,x,f(x_1,\dots,x_n,x)),$$

entonces (omitiendo $x_1,\dots,x_n$) α^* equivale a:

$$\bigvee u(\phi_f(Sx,u) \wedge \bigvee u'(\phi_f(x,u') \wedge \phi_h(x,u',u))),$$

que es un teorema de $\mathrm{I}\Sigma_1$ de nuevo por construcción de ϕ_f.

9. Si α es un axioma de $\mathrm{I}\Sigma_n^+$, entonces α^* es un teorema de $\mathrm{I}\Sigma_n$.

   Sólo falta probarlo para los axiomas enunciados explícitamente en la definición 4.1. Si α es uno de los dos primeros, es un teorema de $\mathcal{L}_a$, luego α^* también, pues en este caso es equivalente a α.

   Finalmente, si α es el principio de inducción correspondiente a la fórmula β de $\mathcal{L}_{\mathrm{arp}}$, es fácil ver que α^* es el caso del principio de inducción correspondiente a β^*, y si β es Σ_n en $\mathcal{L}_{\mathrm{arp}}$, entonces β^* es Σ_n en $\mathcal{L}_a$, luego α^* es un axioma de $\mathrm{I}\Sigma_n$.

Ahora el mismo argumento con el que hemos terminado la prueba de 3.29 nos da aquí también la conclusión. ∎

Así pues, lo que hemos probado es que si f es un funtor n-ádico de $\mathcal{L}_{\mathrm{arp}}$, la fórmula $\phi_f(x_1,\dots,x_n,y)$ define en $\mathrm{I}\Sigma_1$ la misma función recursiva primitiva que determina la definición de f en ARP. Si queremos darle nombre, o bien añadimos a $\mathcal{L}_a$ los funtores de $\mathcal{L}_{\mathrm{arp}}$, con lo que obtenemos una extensión intrascendente en la que disponemos del funtor f para nombrarla (y se cumple $y = f(x_1,\dots,x_n) \leftrightarrow \phi_f(x_1,\dots,x_n,y)$), o bien usamos el descriptor para definir

$$\bar{f}(x_1,\dots,x_n) \equiv y \mid \psi_f(x_1,\dots,x_n,y).$$

En $\mathrm{I}\Sigma_1$ (con descriptor) podemos probar que el término así definido cumple la definición del funtor f.

Pero, tanto si usamos descriptores como funtores, tenemos garantizado que con ello obtenemos una extensión intrascendente de $\mathrm{I}\Sigma_1$, en la que no podemos enunciar nada nuevo ni demostrar nada que antes pudiéramos enunciar, pero no demostrar.

A la vista de estos resultados ya no distinguiremos entre $\mathrm{I}\Sigma_n^+$ o AP^+ y las teorías correspondientes $\mathrm{I}\Sigma_n$ o AP. Esto significa que en $\mathrm{I}\Sigma_n$ y en AP podemos usar nombres para todas las funciones recursivas primitivas. Estos nombres pueden ser interpretados como funtores, como descripciones, o como meras formas de abreviar las fórmulas $\phi_f(y)$.

En particular podemos afirmar que $\mathrm{I}\Sigma_1$ es una extensión de ARP y, más aún, el teorema [CS 1.24] prueba que se trata de una extensión conservativa para fórmulas de tipo Π_2, es decir:

> *Una fórmula de tipo* Π_2 *de* $\mathcal{L}_{\mathrm{arp}}$ *es demostrable en* ARP *si y sólo si es demostrable en* $\mathrm{I}\Sigma_1$.

En particular, la consistencia de $\mathrm{I}\Sigma_1$ equivale a la de ARP.

Si tenemos en cuenta que ARP no es más que un "envoltorio" conveniente de la teoría ARP presentada en el capítulo I, de modo que las fórmulas "genuinas" de ARP (las que se corresponden con fórmulas de dicha presentación sin signos lógicos) son las fórmulas Δ_0, entonces podemos afirmar que $\mathrm{I}\Sigma_1$ es una extensión conservativa de ARP en el sentido de que toda fórmula del lenguaje $\mathcal{L}_{\mathrm{arp}}$ (el original, sin signos lógicos) es demostrable en ARP si y sólo si lo es en $\mathrm{I}\Sigma_1$.

Para referencias posteriores, recapitulamos aquí los resultados que hemos obtenido (c.a. = cuantificadores acotados):

	ARP	ARP^+	$\mathrm{I}\Sigma_1^+$	$\mathrm{I}\Sigma_1$
Lenguaje	$\mathcal{L}_{\mathrm{arp}}$	$\mathcal{L}_{\mathrm{arp}}^+$	$\mathcal{L}_{\mathrm{arp}}^+$	$\mathcal{L}_a$
Fórmulas Δ_0	todas	atómicas	atómicas	c.a.
		$\updownarrow$	$\updownarrow$	
		c.a.	c.a.	
Inducción	todas	Δ_0	Σ_1	Σ_1

1. La aritmética recursiva primitiva (original), está definida sobre el lenguaje $\mathcal{L}_{\mathrm{arp}}$, sin conectores ni cuantificadores, que sólo admite fórmulas atómicas (igualdades), mientras que ARP^+ está definida sobre el lenguaje de primer orden que resulta de añadir a $\mathcal{L}_a$ los funtores de $\mathcal{L}_{\mathrm{arp}}$ (identificando, los funtores "siguiente", $+$ y $\cdot$ de $\mathcal{L}_a$ con los funtores correspondientes de $\mathcal{L}_{\mathrm{arp}}^+$).

2. Cada fórmula de $\mathcal{L}_{\mathrm{arp}}^+$ con cuantificadores acotados es equivalente en ARP^+ a una fórmula atómica (es decir, de $\mathcal{L}_{\mathrm{arp}}$), que es demostrable en ARP^+ si y sólo si lo es en ARP (teorema 4.5).

   De este modo, las fórmulas de $\mathcal{L}_{\mathrm{arp}}$ se corresponden con las fórmulas con cuantificadores acotados de $\mathcal{L}_{\mathrm{arp}}^+$, pero, como de una fórmula siempre podemos deducir la que resulta de cuantificar universalmente sus variables, y viceversa, tenemos que los teoremas de $\mathcal{L}_{\mathrm{arp}}$ se corresponden en realidad con los teoremas de tipo Π_1 de $\mathcal{L}_{\mathrm{arp}}^+$.

3. La teoría $\mathrm{I}\Sigma_1^+$ es la que resulta de extender el principio de inducción de ARP^+ de fórmulas Δ_0 a fórmulas Σ_1.

   Una fórmula de tipo Π_2 de $\mathcal{L}_{\mathrm{arp}}^+$ es demostrable en ARP^+ si y sólo si es demostrable en $\mathrm{I}\Sigma_1^+$ [CS 1.24]. En particular, una fórmula de $\mathcal{L}_{\mathrm{arp}}$ es demostrable en ARP si y sólo si lo es en $\mathrm{I}\Sigma_1^+$.

4. A cada fórmula de $\mathcal{L}_{\mathrm{arp}}^+$ le podemos asociar una fórmula de $\mathcal{L}_a$ equivalente a ella en $\mathrm{I}\Sigma_1^+$ (del mismo tipo Σ_n o Π_n, pero la traducción de una fórmula Δ_0 es, en general, Δ_1), que es demostrable en $\mathrm{I}\Sigma_1^+$ si y sólo si su traducción lo es en $\mathrm{I}\Sigma_1$ (teorema 4.49). En particular, los teoremas de ARP se corresponden con los teoremas de tipo Π_1 de $\mathrm{I}\Sigma_1$.

Una vez establecidos estas relaciones fundamentales, llamamos ARP a la teoría axiomática de primer orden que hemos construido como ARP^+ y llamamos $\mathrm{I}\Sigma_1$ indistintamente a la teoría que hemos definido como tal o a la extensión intrascendente $\mathrm{I}\Sigma_1^+$, pues no hay diferencia entre añadir a $\mathrm{I}\Sigma_1$ los funtores de $\mathcal{L}_{\mathrm{arp}}$ (con sus definiciones) o definirlos como descripciones sin añadir más signos a $\mathcal{L}_a$.

Así pues, en lo sucesivo, no hay razón para trabajar en ARP, cuando podemos trabajar en $\mathrm{I}\Sigma_1$, es decir, admitiendo el principio de inducción para fórmulas de tipo Σ_1 o Π_1, y no sólo Δ_0, como hasta ahora. Cabe destacar el teorema [CS 1.23], según el cual, si en $\mathrm{I}\Sigma_1$ demostramos un resultado de la forma $\bigvee v\,\phi(x_1,\ldots,x_n,v)$ (donde ϕ es Σ_1), podemos asegurar que existe un funtor F de ARP que cumple $\phi(x_1,\ldots,x_n,F(x_1,\ldots,x_n))$, de modo que la prueba de la existencia es constructiva, aunque pueda no parecerlo.

De aquí se deduce una consecuencia notable:

Teorema 4.50 *Toda fórmula de tipo $\Delta_1^{\mathrm{I}\Sigma_1}$ es equivalente en $\mathrm{I}\Sigma_1$ a una fórmula de tipo Δ_0 de $\mathcal{L}_{\mathrm{arp}}$ (es decir, atómica).*

Demostración: Sea $\alpha(x_1,\ldots,x_n)$ una fórmula de $\mathcal{L}_{\mathrm{arp}}$ de tipo $\Delta_1^{\mathrm{I}\Sigma_1}$, es decir, tal que existen fórmulas $\phi(x_1,\ldots,x_n,x)$, $\psi(x_1,\ldots,x_n,x)$ de tipo Δ_0 en $\mathcal{L}_{\mathrm{arp}}$ (es decir, atómicas[29]) tales que en $\mathrm{I}\Sigma_1$ se demuestra:

$$\alpha(x_1,\ldots,x_n) \leftrightarrow \bigvee u\,\phi(x_1,\ldots,x_n,u) \leftrightarrow \neg\bigvee u\,\psi(x_1,\ldots,x_n,u).$$

Entonces en $\mathrm{I}\Sigma_1$ se demuestra también que

$$\bigvee u(\phi(x_1,\ldots,x_n,u) \vee \psi(x_1,\ldots,x_n,u)).$$

Por [CS 1.23] existe un funtor n-ádico F de $\mathcal{L}_{\mathrm{arp}}$ tal que en ARP se demuestra

$$\phi(x_1,\ldots,x_n,F(x_1,\ldots,x_n)) \vee \psi(x_1,\ldots,x_n,F(x_1,\ldots,x_n)).$$

[29]Si consideramos, en principio a $\mathrm{I}\Sigma_1$ como teoría sobre $\mathcal{L}_a$, las fórmulas ϕ y ψ son Δ_0 en $\mathcal{L}_a$, es decir, que tienen cuantificadores acotados, pero por 4.5 podemos sustituirlas por fórmulas atómicas de $\mathcal{L}_{\mathrm{arp}}$.

Por lo tanto, en $\mathrm{I}\Sigma_1$ se cumple que

$$\alpha(x_1, \ldots, x_n) \leftrightarrow \phi(x_1, \ldots, x_n, F(x_1, \ldots, x_n)).$$

y la fórmula $\phi(x_1, \ldots, x_n, F(x_1, \ldots, x_n))$ es atómica. ∎

Así, mientras en $\mathrm{I}\Sigma_1$ (sin funtores añadidos) las fórmulas Δ_1 son más generales que las fórmulas Δ_0, acabamos de ver que, al añadir los funtores de $\mathcal{L}_{\mathrm{arp}}$, las fórmulas Δ_1 en sentido amplio son las mismas que las Δ_0 en sentido amplio.

Capítulo V

Teorías de conjuntos

Puesto que $\mathrm{I}\Sigma_1$ extiende a ARP, al razonar en $\mathrm{I}\Sigma_1$ podemos usar todos los conceptos que tenemos definidos en ARP y todos los resultados que tenemos demostrados sobre ellos. En particular contamos con la relación de pertenencia $x \in y$ que tenemos definida en ARP, que es una fórmula atómica de $\mathcal{L}_{\mathrm{arp}}$ equivalente a una fórmula Δ_1 de $\mathcal{L}_a$. De hecho, la hemos definido explícitamente en la nota posterior a la definición 4.46.

Ahora bien, a la hora de emplear en $\mathrm{I}\Sigma_1$ los resultados sobre conjuntos probados en ARP hay que tener en cuenta que "para toda fórmula de $\mathcal{L}_{\mathrm{arp}}$" no significa lo mismo que "para toda fórmula de $\mathcal{L}_a$" (ni de $\mathcal{L}^+_{\mathrm{arp}}$), por lo que tenemos que ser conscientes del alcance real de los resultados que tenemos demostrados. En este capítulo vamos a hacer más que eso: vamos a axiomatizar los resultados conjuntistas que podemos probar en $\mathrm{I}\Sigma_1$ mediante teorías axiomáticas independientes, pero estrechamente relacionadas con $\mathrm{I}\Sigma_1$.

Podemos considerar que todos los conceptos y resultados de este capítulo se definen y demuestran en ARP (aunque, según hemos visto en el capítulo anterior, en la práctica podemos trabajar en $\mathrm{I}\Sigma_1$).

5.1 La teoría básica de conjuntos

Definición 5.1 El *lenguaje de la teoría de conjuntos* $\mathcal{L}_{\mathrm{tc}}$ es el lenguaje formal (con descriptor) cuyo único signo eventual es un relator diádico que representaremos por $\in$ y lo llamaremos *relator de pertenencia*. Escribiremos $t_1 \notin t_2 \equiv \neg t_1 \in t_2$. También es frecuente abreviar

$$\bigwedge u \in x\, \alpha \equiv \bigwedge u(u \in x \to \alpha), \qquad \bigvee u \in x\, \alpha \equiv \bigvee u(u \in x \wedge \alpha).$$

La *teoría básica de conjuntos* es la teoría axiomática B cuyos axiomas son:

Extensionalidad	$\bigwedge xy(\bigwedge u(u \in x \leftrightarrow u \in y) \to x = y)$
Par	$\bigwedge xy\bigvee z\bigwedge u(u \in z \leftrightarrow u = x \vee u = y)$
Unión	$\bigwedge x\bigvee z\bigwedge u(u \in z \leftrightarrow \bigvee v(u \in v \wedge v \in x))$
Diferencia	$\bigwedge xy\bigvee z\bigwedge u(u \in z \leftrightarrow u \in x \wedge u \notin y)$

El lector observará que todos los axiomas de B son teoremas de ARP si interpretamos que $x \in y$ no es la fórmula atómica de $\mathcal{L}_{\mathrm{tc}}$ que de hecho es, sino la fórmula que hemos definido en ARP. Por lo tanto, podemos considerar que las fórmulas de $\mathcal{L}_{\mathrm{tc}}$ hablan de los mismos conjuntos de los que podemos hablar en ARP o en $\mathrm{I}\Sigma_1$, pero desde otro lenguaje formal. Más precisamente, puesto que todos los axiomas de B pueden demostrarse en ARP, de hecho, al demostrar teoremas en B, estaremos demostrando teoremas de ARP. Más adelante precisaremos estas ideas y las discutiremos con detalle, pero de momento vamos a analizar un poco la teoría B.

En primer lugar, puesto que los axiomas de B no tienen descriptores, podemos considerar por un momento la teoría con los mismos axiomas, pero sobre el lenguaje que resulta de quitarle a $\mathcal{L}_{\mathrm{tc}}$ el descriptor. Aplicando EG al cuarto axioma obtenemos

$$\bigvee z \bigwedge u(u \in z \leftrightarrow u \in x \wedge u \notin x),$$

de donde se sigue $\bigvee z \bigwedge u\, u \notin x$, es decir, vemos que existe un conjunto vacío, un conjunto sin elementos. Pero el axioma de extensionalidad implica que dicho conjunto es único, luego, si llamamos $\phi(x) \equiv \bigwedge u\, u \notin x$, hemos probado $\overset{1}{\bigvee} u\, \phi(u)$. Esto nos permite aplicar el teorema 3.29 para concluir que la teoría que resulta de añadir el descriptor (es decir, B) junto con el axioma

$$\bigwedge u\, u \notin v | v = v,$$

es una extensión intrascendente. En particular, podemos considerar que esta fórmula es un axioma de B, lo cual no es más que el convenio de considerar que cualquier descripción impropia hace referencia al conjunto vacío. A partir de aquí (una vez probado que $\overset{1}{\bigvee} u\, \phi(u)$ se demuestra sin descriptores) ya podemos trabajar en B con descriptores sabiendo que todo teorema de B tiene un equivalente sin descriptores demostrable en B sin descriptores.

En particular tenemos que en B se prueba $\overset{1}{\bigvee} v \bigwedge u\, u \notin v$, lo que nos permite definir el *conjunto vacío* como

$$\varnothing \equiv v | \bigwedge u\, u \notin v,$$

y la regla de las descripciones propias nos da que $\bigwedge u\, u \notin \varnothing$. Más aún, la unicidad nos da que

$$\bigwedge u\, u \notin x \to x = \varnothing.$$

En estos términos, hemos introducido el axioma-convenio $v|(v = v) = \varnothing$.

El axioma de extensionalidad implica que los conjuntos z cuya existencia afirman los otros tres axiomas son únicos, lo cual nos permite definir:

$$\begin{aligned} \{x, y\} &\equiv v | \bigwedge u(u \in w \leftrightarrow u = x \vee u = y), \\ \bigcup x &\equiv w | \bigwedge u(u \in w \leftrightarrow \bigvee v(u \in v \wedge v \in x)), \\ x \setminus y &\equiv w | \bigwedge u(u \in w \leftrightarrow u \in x \wedge u \notin y), \end{aligned}$$

y la regla de las descripciones propias nos da los teoremas:

$$u \in \{x, y\} \leftrightarrow u = x \vee u = y,$$

$$u \in \bigcup x \leftrightarrow \bigvee v(u \in v \wedge v \in x),$$

$$u \in x \setminus y \leftrightarrow u \in x \wedge y \notin y.$$

Más aún, en B podemos demostrar:

$$\overset{1}{\bigvee} z \bigwedge u(u \in z \leftrightarrow u \in x \vee u \in y).$$

En efecto, la unicidad la proporciona el axioma de extensionalidad, mientras que la existencia se prueba considerando $\bigcup\{x, y\}$. Esto nos permite definir

$$x \cup y \equiv z|\bigwedge u(u \in z \leftrightarrow u \in x \vee u \in y)$$

y resulta que

$$u \in x \cup y \leftrightarrow u \in x \vee u \in y.$$

A su vez,

$$\overset{1}{\bigvee} z \bigwedge u(u \in z \leftrightarrow u \in x \wedge u \in y),$$

pues la existencia nos la da $z = (x \cup y) \setminus ((x \setminus y) \cup (y \setminus x))$ y la unicidad el axioma de extensionalidad. Por lo tanto, podemos definir

$$x \cap y \equiv z|\bigwedge u(u \in z \leftrightarrow u \in x \wedge u \in y)$$

y demostrar:

$$u \in x \cap y \leftrightarrow u \in x \wedge u \in y.$$

Otro concepto conjuntista básico que tenemos a nuestra disposición en B es el de *par ordenado:*

$$(x, y) \equiv \{\{x\}, \{x, y\}\}.$$

Es pura rutina comprobar que

$$(x, y) = (x', y') \leftrightarrow x = x' \wedge y = y'.$$

Por último introducimos la inclusión:

$$x \subset y \equiv \bigwedge u(u \in x \to u \in y),$$

sobre la cual podemos probar fácilmente:

$$x \subset x, \qquad x \subset y \wedge y \subset x \to x = y, \qquad x \subset y \wedge y \subset z \to x \subset z.$$

Clases y conjuntos Hasta aquí hemos analizado el contenido de los axiomas de B, pero antes de seguir extrayendo consecuencias, conviene introducir unos convenios generales de notación.

Si $\phi(x)$ es cualquier fórmula de $\mathcal{L}_{\text{tc}}$ (tal vez con más variables libres), escribiremos

$$A \equiv \{x \mid \phi(x)\}$$

(y leeremos que "A es la *clase* de todos los conjuntos que cumplen $\phi(x)$") para indicar que, en lo sucesivo, siempre que escribamos $x \in A$ habrá que entenderlo como $\phi(x)$. Más precisamente, si $B = \{x \mid \psi(x)\}$ es otra clase, entenderemos que:

1. $x \in A \equiv \phi(x)$.
2. $y = A \equiv \bigwedge x(x \in y \leftrightarrow x \in A) \equiv \bigwedge x(x \in y \leftrightarrow \phi(x))$.
3. $A = B \equiv \bigwedge x(x \in A \leftrightarrow x \in B) \equiv \bigwedge x(\phi(x) \leftrightarrow \psi(x))$.
4. $A \in z \equiv \bigvee y \in z\ y = A \equiv \bigvee y \in z \bigwedge x(x \in y \leftrightarrow \phi(x))$.
5. $A \in B \equiv \bigvee y(y = A \wedge y \in B) \equiv \bigvee y(\bigwedge x(x \in y \leftrightarrow \phi(x)) \wedge \psi(y))$.

Notemos que si una fórmula contiene una variable, ésta aparece necesariamente en subfórmulas de tipo $x = y$ o $x \in y$, por lo que si en cualquier fórmula de $\mathcal{L}_{\text{tc}}$ sustituimos algunas de sus variables libres por una o varias clases, la expresión resultante nombra a una fórmula concreta de $\mathcal{L}_{\text{tc}}$, la que resulta de sustituir las subfórmulas atómicas que contienen clases por las definiciones que acabamos de dar.

Por ejemplo, si escribimos $A \subset B$, no hay ninguna duda sobre a qué fórmula de $\mathcal{L}_{\text{tc}}$ nos estamos refiriendo: según la definición de la inclusión se trata de la fórmula

$$\bigwedge x(x \in A \to x \in B) \equiv \bigwedge x(\phi(x) \to \psi(x)).$$

Para interpretar un término t definido en términos de clases aplicaremos el criterio anterior a una fórmula equivalente[1] a $x \in t$ que no tenga descriptores. Por ejemplo, para interpretar qué es $A \cup B$ observamos que

$$x \in y \cup z \leftrightarrow x \in y \vee x \in z,$$

y la fórmula de la derecha ya no tiene descriptores, por lo que interpretamos

$$x \in A \cup B \leftrightarrow x \in A \vee x \in B \leftrightarrow \phi(x) \vee \psi(x).$$

Podemos expresar esto diciendo que

$$A \cup B \equiv \{x \mid x \in A \vee x \in B\}.$$

En resumen: cada clase está definida a partir de una fórmula, y una fórmula que contiene clases se puede interpretar inequívocamente como una fórmula "normal" de $\mathcal{L}_{\text{tc}}$ sin más que sustituir cada aparición de una clase en una subfórmula atómica por la fórmula que hemos indicado. En particular, cuando decimos que dos clases son iguales ($A = B$) esto simplemente significa que las fórmulas que las definen son equivalentes (lo cual puede depender de la teoría axiomática considerada, es decir, que, por ejemplo, puede ocurrir que dos clases sean iguales en una extensión de B y no lo sean en otra).

[1] A partir de aquí lo que hacemos depende de la teoría de conjuntos concreta que consideremos. Todo cuanto decimos vale para cualquier extensión de B.

Si tenemos una clase $A = \{x \mid \phi(x)\}$, diremos que *"A es un conjunto"* para referirnos a la fórmula de $\mathcal{L}_{\mathrm{tc}}$

$$\bigvee y \bigwedge x(x \in y \leftrightarrow x \in A) \equiv \bigvee y \bigwedge x(x \in y \leftrightarrow \phi(x)),$$

y en tal caso identificaremos la clase A con el conjunto y, que será único por el axioma de extensionalidad. En otras palabras, cuando decimos que una clase (definida por una fórmula $\phi(x)$) es un conjunto, queremos decir que existe un conjunto cuyos elementos son justamente los conjuntos que cumplen $\phi(x)$.

Si una clase A no es un conjunto diremos que es una *clase propia.* Concretamente, esto significa que no existe ningún conjunto cuyos elementos sean todos los conjuntos que satisfacen la fórmula que define la clase.

Hay que enfatizar que todos estos convenios hacen que cualquier afirmación que hagamos en lo sucesivo sobre clases en una teoría de conjuntos se pueda entender inequívocamente como una afirmación sobre si una fórmula determinada de $\mathcal{L}_{\mathrm{tc}}$ es o no un teorema.

También cabe destacar que los convenios precedentes no pueden justificar de ningún modo una cuantificaciín sobre clases, es decir, no podemos expresar mediante una fórmula de $\mathcal{L}_{\mathrm{tc}}$ una afirmación del tipo "para toda clase A" o "existe una clase A".

Ejemplo: La paradoja de Russell Consideremos la *clase de Russell,* definida como

$$R \equiv \{x \mid x \notin x\},$$

es decir, la clase de todos los conjuntos que no se pertenecen a sí mismos. Podemos probar en B que R no es un conjunto. Según los convenios que hemos adoptado, "R es un conjunto" es una forma de nombrar la fórmula siguiente:

$$\bigvee y \bigwedge x(x \in y \leftrightarrow x \notin x),$$

pero podemos probar la negación de esta fórmula. Por reducción al absurdo, suponemos que existe un y que cumple $\bigwedge x(x \in y \leftrightarrow x \notin x)$. Entonces, en particular, $y \in y \leftrightarrow y \notin y$, pero esta fórmula es contradictoria. ∎

Otro ejemplo de clase es la *clase universal:*

$$V \equiv \{x \mid x = x\}$$

o *la clase de todos los conjuntos.* La teoría B no es lo suficientemente potente como para decidir si V es o no un conjunto, pero en las extensiones de B mas habituales puede probarse que no lo es.

Ordinales A pesar de su simplicidad, la teoría B permite definir los números naturales. La idea básica es que podemos definir

$$x' \equiv x \cup \{x\}$$

Con esta definición de "sucesor" podemos ir definiendo

$$0 \equiv \varnothing, \quad 1 \equiv 0' = 0 \cup \{0\} = \{0\}, \quad 2 \equiv 1' = 1 \cup \{1\} = \{0, 1\}, \quad 3 \equiv 2' = \{0, 1, 2\}$$

y así sucesivamente. El problema es que "y así sucesivamente" no sirve. Necesitamos definir una fórmula de $\mathcal{L}_{\mathrm{tc}}$ que podamos toma como definición de "número natural" en la teoría B. Para ello empezamos definiendo algunos conceptos:

Definición 5.2 Diremos que un conjunto x es

1. *transitivo* si $\bigwedge u \in x\ u \subset x$,
2. $\in$*-conexo* si $\bigwedge uv \in x(u \in v \vee v \in u \vee u = v)$,
3. *bien fundado* si $\bigwedge u(u \subset x \wedge u \neq \varnothing \to \bigvee v \in u\, v \cap u = \varnothing)$. Un conjunto v que cumpla esta definición recibe el nombre de *elemento* $\in$*-minimal* de u.
4. x es un *ordinal* si cumple las tres propiedades anteriores.

Llamaremos Ω a la clase de todos los ordinales (es decir, que $x \in \Omega \equiv x$ es un ordinal).

Con más detalle: un conjunto x es transitivo si todos sus elementos son también subconjuntos suyos. Una forma equivalente de expresar esta propiedad es $\bigwedge uv(u \in v \wedge v \in x \to u \in x)$, y de ahí el nombre de "transitividad" (de la pertenencia).

Un conjunto es $\in$-conexo si dos cualesquiera de sus elementos están "conectados" por la pertenencia, en el sentido de que uno pertenece al otro (entendiendo que hablamos de dos elementos distintos). Notemos que si y es $\in$-conexo y $x \subset y$ entonces x también es $\in$-conexo, pues dos de sus elementos son también elementos de y, luego están conectados por la pertenencia.

Por último, un conjunto x está bien fundado si cada subconjunto no vacío u tiene un elemento $\in$-minimal, es decir un $v \in u$ tal que ningún elemento $w \in u$ pertenece a v. También se cumple que si y está bien fundado y $x \subset y$ entonces x está bien fundado, pues todo $u \subset x$ no vacío cumple también $u \subset y$, luego tiene un $\in$-minimal por la buena fundación de y. Necesitaremos una propiedad elemental de los conjuntos bien fundados:

Teorema 5.3 *Si x es un conjunto bien fundado entonces $x \notin x$.*

Demostración: Si $x \in x$ entonces $\{x\} \subset x \wedge \{x\} \neq \varnothing$. Sea u un elemento $\in$-minimal de $\{x\}$. Necesariamente, $u = x$, pero $x \in x \cap \{x\}$, contradicción. ∎

Recordemos que hemos definido $0 \equiv \varnothing$ y $x' \equiv x \cup \{x\}$. Ahora podemos probar:

Teorema 5.4 $0 \in \Omega \wedge \bigwedge x \in \Omega\ x' \in \Omega$.

DEMOSTRACIÓN: Notemos que $0 = \varnothing$ cumple trivialmente las tres condiciones de la definición de ordinal (es transitivo porque no existe ningún $u \in \varnothing$ que pueda incumplir la definición, es $\in$-conexo porque no existen $u, v \in \varnothing$ que puedan incumplir la definición, y está bien fundado porque no existe ningún $u \subset \varnothing$, $u \neq \varnothing$ que pueda incumplir la definición).

Supongamos ahora que x es un ordinal. Si $u \in x' = x \cup \{x\}$, entonces $u \in x \vee u = x$, pero en ambos casos $u \subset x$, en el primero porque x es transitivo. Esto prueba que x' es transitivo.

Si $u, v \in x'$, entonces $u \in x \vee u = x$ y $v \in x \vee v = x$. Esto nos da cuatro casos: $u \in x \wedge v \in x$ o bien $u \in x \wedge v = x$, o bien $u = x \wedge v \in x$, o bien $u = x = v$. En el primero tenemos que $u \in v \vee v \in u \vee u = v$ porque x es $\in$-conexo, y en los otros tres tenemos $u \in v$, $v \in u$, $u = v$ respectivamente. Esto prueba que x' es $\in$-conexo.

Tomemos $u \subset x' \wedge u \neq \varnothing$ y veamos que tiene $\in$-minimal. Tratemos aparte el caso en que $u = \{x\}$. Entonces $v = x$ es un $\in$-minimal de u, pues $x \cap \{x\} = \varnothing$. En efecto, si existiera $w \in x \cap \{x\}$, sería $x = w \in x$, en contradicción con el teorema anterior.

Como $u \subset x \cup \{x\}$, si no se da la igualdad $u = \{x\}$ es porque $u \cap x \neq \varnothing$, y tenemos así un subconjunto no vacío de x. Como x está bien fundado existe un $v \in u \cap x$ que es $\in$-minimal para esta intersección. Vamos a ver que es $\in$-minimal de u.

En efecto, si $w \in v \cap u$, entonces $w \in x'$, luego $w \in x \vee w = x$. En el primer caso $w \in u \cap x$ y $w \in v$, lo que contradice que v sea $\in$-minimal de $u \cap x$. En el segundo caso $x = w \in v \in x$, luego, por la transitividad de x, resulta que $x \in x$, en contradicción con el teorema anterior. ∎

En vista de este teorema resulta que 0 es un ordinal, luego $1 = 0'$ es un ordinal, luego $2 = 1'$ es un ordinal y, en definitiva, todos los números naturales son ordinales, pero no podemos demostrar tal cosa porque no tenemos una definición de número natural. En realidad vamos a definir los números naturales como los ordinales que cumplen una propiedad adicional, pero antes de ello necesitamos demostrar algunas propiedades sobre ordinales.

Notemos que, por ejemplo, $5 = \{0, 1, 2, 3, 4\}$, de modo que los elementos del ordinal 5 son otros ordinales. Esto es cierto en general:

Teorema 5.5 *Los elementos de los ordinales son ordinales.*

DEMOSTRACIÓN: Sea $y \in \Omega$ y sea $x \in y$. Por transitividad $x \subset y$ y por consiguiente x es conexo y bien fundado. Falta probar que es transitivo, es decir, que $\bigwedge uv(u \in v \wedge v \in x \to u \in x)$.

Si $u \in v \wedge v \in x$, tenemos $v \in x \wedge x \in y$, y como y es transitivo, $v \in y$, e igualmente $u \in y$. Así pues, $\{u, v, x\} \subset y$. Como y está bien fundado se cumplirá

$$u \cap \{u, v, x\} = \varnothing \quad \vee \quad v \cap \{u, v, x\} = \varnothing \quad \vee \quad x \cap \{u, v, x\} = \varnothing,$$

pero $u \in v \cap \{u, v, x\}$ y $v \in x \cap \{u, v, x\}$, luego ha de ser $u \cap \{u, v, x\} = \varnothing$. Como y es conexo ha de ser $u \in x \vee x \in u \vee u = x$, pero si $x \in u$ entonces $x \in u \cap \{u, v, x\} = \varnothing$, y si $x = u$ entonces $v \in u \cap \{u, v, x\} = \varnothing$. Así pues, se ha de cumplir $u \in x$, como queríamos. ∎

Notemos que el teorema anterior puede enunciarse así:

$$\bigwedge xy(x \in y \wedge y \in \Omega \to x \in \Omega),$$

pero esto es lo mismo que decir que la clase Ω es transitiva.

Definición 5.6 Si x, y son ordinales, escribiremos $x \leq y \equiv x \subset y$.

Es inmediato que se cumplen los teoremas siguientes son válidos para conjuntos arbitrarios:

1. $x \leq x$,
2. $x \leq y \wedge y \leq x \to x = y$,
3. $x \leq y \wedge y \leq z \to x \leq z$.

y esto se expresa diciendo que $\leq$ (o, lo que es lo mismo, la inclusión) es una relación de orden parcial, pero cuando nos restringimos a ordinales tenemos más:

$$x \in \Omega \wedge y \in \Omega \to x \leq y \vee y \leq x.$$

En efecto, si $x \in \Omega$ y $u, v \in x$, entonces $u \in v \vee v \in u \vee u = v$, y como sabemos que u, v son ordinales, en particular son transitivos, y resulta que $u \subset v \vee v \subset u$. Más aún:

Teorema 5.7 *Si x es un ordinal y $\varnothing \neq u \subset x$, entonces todo $\in$-minimal v de u es, de hecho, el mínimo de u para la relación de inclusión, es decir:*

$$\bigwedge w \in u\, v \leq w.$$

Demostración: En principio tenemos que $v \cap u = \varnothing$. Si $w \in u \subset x$, entonces $v, w \in x$, luego $w \in v \vee v \in w \vee v = w$. El caso $w \in v$ no puede darse, pues implicaría que $w \in v \cap u = \varnothing$. En los otros dos casos, como w es un ordinal (por ser elemento de x) es transitivo y se cumple que $v \subset w$, es decir, $v \leq w$. ∎

Así pues, todo conjunto no vacío contenido en un ordinal tiene un mínimo elemento. Esto se expresa diciendo que los ordinales están bien ordenados por la inclusión.

Notemos que el mínimo es necesariamente único, por lo que hemos probado que cada subconjunto no vacío de un ordinal tiene un único $\in$-minimal.

Cada número natural (en el sentido en que pensamos definirlos en B) está formado por los números menores que él. Así, el hecho de que (informalmente) $3 < 5$, se traduce en B en que $3 \in 5$. Para sacarle partido a esta observación demostramos lo siguiente:

Teorema 5.8 $\bigwedge xy \in \Omega(x \leq y \to x \in y \vee x = y)$.

DEMOSTRACIÓN: Si $x \neq y$ entonces $y \setminus x \neq \varnothing$. Como y es un ordinal, $y \setminus x$ tiene un elemento minimal $u \in y \setminus x$, de modo que $u \cap (y \setminus x) = \varnothing$. Basta probar que $u = x$, pues entonces tenemos que $x \in y$.

Si $z \in u$, entonces $z \notin y \setminus x$ y $z \in y$ (por transitividad, pues $z \in u \in y$), luego $z \in x$. Por lo tanto $u \subset x$.

Si $z \in x$, entonces tenemos z, $u \in y$, luego $z \in u \vee u \in z \vee z = u$. Si $u \in z$, entonces $u \in z \in x$, luego $u \in x$, contradicción ($u \in y \setminus x$). Si $z = u$ entonces de nuevo $u \in x$, contradicción. Por lo tanto $z \in u$, y así $x \subset u$. En definitiva, tenemos la igualdad $u = x$. ∎

Notemos que no pueden darse a la vez los dos casos $x \in y \wedge x = y$, pues esto supondría que $y \in y$, cuando hemos probado que un conjunto bien fundado no se pertenece a sí mismo. Por lo tanto, si definimos

$$x < y \equiv x \leq y \wedge x \neq y,$$

tenemos que si x, y son ordinales entonces $x < y \leftrightarrow x \in y$.

Teorema 5.9 *La intersección de dos ordinales es un ordinal.*

DEMOSTRACIÓN: Sean x, y ordinales. Como $x \cap y \subset x$, trivialmente $x \cap y$ es conexo y bien fundado. Falta ver que es transitivo. En efecto: si $u \in x \cap y$, entonces $u \in x \wedge u \in y$, $u \subset x \wedge u \subset y$, luego $u \subset x \cap y$. ∎

Con esto podemos probar un resultado no trivial. Hasta ahora sabemos que si x y y son dos elementos de un ordinal (dos ordinales que pertenecen a otro ordinal), entonces $x \in y \vee y \in x \vee x = y$. Ahora podemos probar que esto es cierto sin suponer que x e y son elementos de otro ordinal:

Teorema 5.10 $\bigwedge xy \in \Omega\,(x \in y \vee y \in x \vee x = y)$.

DEMOSTRACIÓN: $x \cap y$ es un ordinal, $x \cap y \subset x$ y $x \cap y \subset y$. Por el teorema 5.8 tenemos $(x \cap y \in x \vee x \cap y = x) \wedge (x \cap y \in y \vee x \cap y = y)$. Esto nos da cuatro casos:

$$(x \cap y \in x \wedge x \cap y \in y) \vee (x \cap y \in x \wedge x \cap y = y)$$

$$\vee\, (x \cap y = x \wedge x \cap y \in y) \vee (x \cap y = x \wedge x \cap y = y),$$

o sea $x \cap y \in x \cap y \quad \vee \quad y \in x \quad \vee \quad x \in y \quad \vee \quad x = y$. El primer caso se descarta por el teorema 5.3. ∎

Notemos que el teorema anterior es lo que significa precisamente la sentencia "Ω es una clase $\in$-conexa". También hemos visto que es transitiva, luego estamos a un paso de probar lo siguiente:

Teorema 5.11 Ω *es un ordinal.*

DEMOSTRACIÓN: Sólo nos falta probar que Ω está bien fundada. Si tomamos la definición de "x está bien fundada" y sustituimos x por Ω vemos que lo que tenemos que probar es que

$$\bigwedge u(u \subset \Omega \wedge u \neq \varnothing \to \bigvee v \in u\, v \cap u = \varnothing),$$

donde a su vez hay que entender que $u \subset \Omega$ significa $\bigwedge w(w \in u \to w \in \Omega)$.

En definitiva, hemos de demostrar que si u es un conjunto no vacío cuyos elementos son ordinales, entonces tiene un $\in$-minimal. Podemos tomar $w \in u$ (que será entonces un ordinal) y distinguimos dos casos. Si w es un $\in$-minimal de u, no hay nada que probar. En caso contrario, existe un $z \in w \cap u$, luego $w \cap u \neq \varnothing$. Así $w \cap u$ es un subconjunto no vacío de w, que es un ordinal, luego existe $v \in w \cap u$ que es $\in$-minimal para la intersección. Basta probar que v es $\in$-minimal para u.

En caso contrario existiría un $z \in v \cap u$, pero $z \in v \in w$ y w es transitivo, luego $z \in w$, luego $z \in v \cap (w \cap u)$, en contradicción con la $\in$-minimalidad de v en $w \cap u$. ∎

De aquí podemos deducir un hecho interesante:

Teorema 5.12 *Ω es una clase propia.*

DEMOSTRACIÓN: El enunciado significa que no existe ningún conjunto cuyos elementos sean todos los ordinales. La demostración consiste en observar que si x fuera tal conjunto, toda la prueba del teorema anterior valdría para concluir que x es un ordinal, es decir, que $x \in \Omega$, y entonces debería cumplirse $x \in x$, en contradicción con el teorema 5.3. ∎

Antes de pasar a definir los números naturales demostramos un último teorema sobre ordinales. En lo sucesivo, cuando digamos que α (o cualquier otra letra griega) es un ordinal se entenderá que α es un conjunto que es un ordinal.

Teorema 5.13 *Se cumple:*

1. *0 es el mínimo ordinal.*
2. *Si α es un ordinal, entonces α' también lo es, y es el mínimo ordinal mayor que α (es decir, $\bigwedge \beta \in \Omega(\alpha < \beta \to \alpha' \leq \beta)$.*
3. *Todo conjunto de ordinales $x \subset \Omega$ tiene supremo[2] $\sigma = \bigcup x$.*

DEMOSTRACIÓN: 1) ya hemos probado que 0 es un ordinal, y es el mínimo porque el conjunto vacío está contenido en cualquier conjunto.

2) Ya hemos probado que $\alpha' \in \Omega$. Si $\alpha < \beta$ entonces $\alpha \in \beta$, luego $\alpha \subset \beta$, luego $\alpha' = \alpha \cup \{\alpha\} \subset \beta$, luego $\alpha' \leq \beta$.

3) Como todo $\alpha \in x$ está contenido en Ω, es claro que $\sigma \subset \Omega$, luego es un conjunto conexo y bien fundado. Hemos de probar que es transitivo, pero

[2]Esto significa que σ es el menor ordinal mayor o igual que todos los elementos de x.

si $\beta \in \sigma$, entonces existe un $\alpha \in x$ tal que $\beta \in \alpha$, luego por la transitividad de α es $\beta \subset \alpha \subset \sigma$. Por consiguiente $\sigma \in \Omega$. Teniendo en cuenta que el orden es la inclusión, es inmediato que σ es el supremo de x. ∎

Definición 5.14 Un conjunto n es un *número natural* si cumple:

$$n \in \Omega \wedge \bigwedge\alpha(\alpha \in n' \to \alpha = 0 \vee \bigvee\beta \in \alpha\, \alpha = \beta').$$

o sea, un número natural n es un ordinal tal que todos los ordinales $0 < \alpha \leq n$ tienen un inmediato anterior (es decir, son el siguiente de otro).

Llamaremos ω a la clase de todos los números naturales (lo que significa que $x \in \omega \equiv x$ es un número natural).

Teorema 5.15 *ω es un ordinal (aunque no necesariamente un conjunto).*

Demostración: Como $\omega \subset \Omega$, es trivialmente una clase $\in$-conexa y bien fundada, y basta ver que es transitiva. Si $u \in v \wedge v \in \omega$, entonces v es un número natural. Por definición tenemos que

$$\bigwedge\alpha(\alpha \in v' \to \alpha = 0 \vee \bigvee\beta \in \alpha\, \alpha = \beta'),$$

como $u < v$, se cumple que $u' \leq v < v'$ y en particular

$$\bigwedge\alpha(\alpha \in u' \to \alpha = 0 \vee \bigvee\beta \in \alpha\, \alpha = \beta'),$$

luego $u \in \omega$. ∎

Teorema 5.16 (Axiomas de Peano) *Se cumple:*

1. $0 \in \omega$,
2. $\bigwedge n \in \omega\, n' \in \omega$,
3. $\bigwedge n \in \omega\, n' \neq 0$,
4. $\bigwedge mn \in \omega(m' = n' \to m = n)$,
5. $\bigwedge y(y \subset \omega \wedge 0 \in y \wedge \bigwedge n \in y\ n' \in y \to y = \omega)$.

Demostración: 1) es trivial.

2) si $n \in \omega$ y $\alpha \in n''$, entonces, o bien $\alpha \in n'$ o bien $\alpha = n'$. En el primer caso $\alpha = 0 \vee \bigvee\beta \in \alpha\, \alpha = \beta'$, porque $n \in \omega$. Esto también se cumple en el segundo caso, tomando $\beta = n$. Por consiguiente $n' \in \omega$.

Las propiedades 3) y 4) son trivialmente válidas para ordinales cualesquiera, pues $0 \leq n < n'$, luego $0 \in n'$, luego $n' \neq 0$. Por otra parte, si $m' = n'$, tiene que ser $m = n$, ya que si fuera $m < n$ entonces $m' \leq n < n'$, luego $m' \neq n'$, e igualmente si $n < m$.

5) Si $y \subset \omega \wedge 0 \in y \wedge \bigwedge n \in y\, n' \in y$ pero $y \neq \omega$, entonces, como hemos probado que Ω es un ordinal, existe un $\in$-minimal $n \in \omega \setminus y$. No puede ser $n = 0$, pues $0 \in y$, $n \notin y$. Como n es un número natural, por definición existe un $m \in n$ tal que $n = m'$. Como n es minimal, no puede ser que $m \in \omega \setminus y$, pues entonces $m \in n \cap (\omega \setminus y)$. Por lo tanto $m \in y$ (notemos que $m \in n \in \omega$, luego $m \in \omega$, por transitividad). Pero estamos suponiendo que $m \in y$ implica $n = m' \in y$, contradicción. ∎

Observemos que la quinta propiedad es una forma del principio de inducción, pero en realidad es muy débil, pues las hipótesis implican que ω es un conjunto, cosa que no puede demostrarse en B, lo que significa que no podemos probar que haya algún conjunto que cumpla la hipótesis.

Poco más se puede decir en B sobre los números naturales. Hemos definido el cero y la operación siguiente, pero no es posible definir la suma o el producto.

Terminamos este apartado mencionando que si ω es un conjunto, entonces $\omega \in \Omega$, y tenemos un ejemplo de ordinal que no es un número natural.

La jerarquía de Lévy Veamos ahora que podemos definir una jerarquía de fórmulas en $\mathcal{L}_{\mathrm{tc}}$ análoga a la jerarquía de Kleene que hemos definido en $\mathcal{L}_a$.

Definición 5.17 Llamaremos semifórmulas Δ_0 de $\mathcal{L}_{\mathrm{tc}}$ a las determinadas por las condiciones siguientes:

1. Las semifórmulas atómicas sin descriptores $x = y$, $x \in y$ son Δ_0.
2. Si α y β son semifórmulas Δ_0, también lo son $\neg\alpha$ y $\alpha \vee \beta$, así como

$$\bigwedge u \in x\, \alpha \equiv \bigwedge u(u \in x \to \alpha), \qquad \bigvee u \in x\, \alpha \equiv \bigvee u(u \in x \wedge \alpha).$$

   donde $u \not\equiv x$.

Las fórmulas Δ_0 son las semifórmulas Δ_0 que además son fórmulas.

Aquí suponemos que los únicos conectores de $\mathcal{L}_{\mathrm{tc}}$ son el negador y el disyuntor. En caso contrario incluimos los restantes en la definición, de modo que, en cualquier caso, si α y β son Δ_0, también lo son $\alpha \to \beta$, $\alpha \wedge \beta$ y $\alpha \leftrightarrow \beta$.

Más llanamente, llamaremos fórmulas Δ_0 a las fórmulas sin descriptores cuyos cuantificadores están todos acotados de la forma indicada.

Diremos que una fórmula de $\mathcal{L}_{\mathrm{tc}}$ es de tipo Σ_n (resp. Π_n), con $n \geq 1$, si consta de una semifórmula Δ_0 precedida a lo sumo n cuantificadores alternados, de modo que, en caso de que haya exactamente n, el primero es un particularizador en el caso de las fórmulas Σ_n o un generalizador en el caso de las fórmulas Π_n.

Más en general, si T es una extensión de B, diremos que una fórmula cualquiera es de tipo Σ_n^T o Π_n^T si es equivalente en T a una fórmula del tipo correspondiente. Una fórmula es de tipo Δ_n^T si es a la vez Σ_n^T y Π_n^T. Normalmente

omitiremos el superíndice T si está claro por el contexto en qué teoría estamos trabajando.[3]

Siempre podemos añadir cuantificadores delante o detrás del prefijo de una fórmula con variables que no aparezcan en ella, y la fórmula resultante es equivalente, por lo que toda fórmula Σ_n o Π_n es Δ_{n+1}.

Un término t es Σ_n, Π_n o Δ_n si lo es la fórmula $x = t$, donde x es una variable que no esté en t. En realidad, todo término Σ_n es Δ_n, pues

$$x = t \leftrightarrow \bigwedge u(u = t \to u = x).$$

Se cumple un teorema análogo a 4.21. Aquí usamos que la fórmula $u = (x, y)$ es Δ_0^{B}. En efecto:

$$u = \{x, y\} \leftrightarrow \bigwedge v \in u\ (v = x \vee v = y) \wedge x \in u \wedge y \in v,$$

$$u = (x, y) \leftrightarrow \bigvee vw \in u\ (v = \{x\} \wedge w = \{x, y\}) \wedge \bigwedge v \in u(u = \{x\} \vee u = \{x, y\}).$$

Además hay que usar que

$$\bigvee xy\ u = (x, y) \leftrightarrow \bigvee v \in u \bigvee xy \in v\ u = (x, y).$$

Teorema 5.18 *Sea T una teoría axiomática que contenga a* B*. Para cada número natural $n \geq 1$ y para fórmulas α y β cualesquiera se cumple:*[4]

1. *Si α, β son Σ_n, lo mismo vale para $\bigvee x\alpha, \alpha \wedge \beta$ y $\alpha \vee \beta$.*
2. *Si α, β son Π_n, lo mismo vale para $\bigwedge x\alpha, \alpha \wedge \beta$ y $\alpha \vee \beta$.*
3. *Si α es Σ_n entonces $\neg\alpha$ es Π_n, y viceversa.*
4. *Si α es Π_n (resp. Σ_n) y β es Σ_n (resp. Π_n), $\alpha \to \beta$ es Σ_n (resp. Π_n).*
5. *Si α y β son Δ_n, también lo son*

$$\neg\alpha, \quad \alpha \wedge \beta, \quad \alpha \vee \beta, \quad \alpha \to \beta, \quad \alpha \leftrightarrow \beta.$$

Demostración: 1) Por simplicidad supondremos $n = 3$. El caso general es formalmente idéntico. Tenemos que $\alpha \leftrightarrow \bigvee x_1 \bigwedge x_2 \bigvee x_3\, \phi$, donde ϕ es Δ_0. Así

$$\bigvee x\alpha \leftrightarrow \bigvee x x_1 \bigwedge x_2 \bigvee x_3\, \phi \leftrightarrow \bigvee w \bigvee z \in w \bigvee x x_1 \in z(w = (x, x_1) \wedge \bigwedge x_2 \bigvee x_3\, \phi)$$

$$\leftrightarrow \bigvee w \bigwedge x_2 \bigvee x_3 \bigvee z \in w \bigvee x x_1 \in z(w = (x, x_1) \wedge \phi).$$

Si $\beta \leftrightarrow \bigvee y_1 \bigwedge y_2 \bigvee y_3\, \psi$, donde ψ es Δ_0 y las variables y_1, y_2, y_3 son distintas de x_1, x_2, x_3, entonces

$$\alpha \wedge \beta \leftrightarrow \bigvee x_1 y_1 \bigwedge x_2 y_2 \bigvee x_3 y_3(\phi \wedge \psi).$$

[3] Como en el caso de la jerarquía de Kleene tenemos que $\alpha \in \Delta_0$ es una fórmula Δ_0 de $\mathcal{L}_{\mathrm{arp}}$, al igual que $\alpha \in \Sigma_n$ y $\alpha \in \Pi_n$ (entendidas en sentido estricto). En sentido amplio, son fórmulas Σ_1, pues afirman la existencia de una fórmula equivalente y de la demostración de la equivalencia.

[4] Véase además el teorema 5.31, más abajo.

Ahora basta aplicar tres veces el caso ya probado y el correspondiente de 2), que se sigue del que hemos probado aplicando 3). Notemos que 3) es inmediato. El caso de $\alpha \vee \beta$ es idéntico.

Como ya hemos señalado, 3) es inmediato, y 2) se sigue de 1) por 3).

4) se sigue de los apartados anteriores porque $\alpha \to \beta$ equivale a $\neg\alpha \vee \beta$ y e) es evidente. ∎

En cualquier teoría que contenga a B podemos hablar de clases propias en el sentido explicado en la sección precedente, de modo que la notación

$$A = \{x \mid \phi(x, x_1, \ldots, x_n)\}$$

significará que $x \in A$ es una abreviatura por $\phi(x, x_1, \ldots, x_n)$. En particular podemos hablar de clases Σ_n, Π_n o Δ_n según que la fórmula $x \in A$ sea del tipo correspondiente. Diremos que una fórmula $\phi(x_1, \ldots, x_n, x)$ define una función $F : A_1 \times \cdots \times A_n \longrightarrow A$ si cumple

$$\bigwedge x_1 \in A_1 \cdots \bigwedge x_n \in A_n \overset{1}{\bigvee} x \in A \; \phi(x_1, \ldots, x_n, x).$$

En tal caso abreviaremos

$$F(x_1, \ldots, x_n) \equiv x|(x_1 \in A_1 \wedge \cdots \wedge x_n \in A_n \wedge \phi(x_1, \ldots, x_n, x)).$$

Se cumple que si las clases A_i son Δ_n y F es Σ_n (en el sentido de que lo es ϕ), con $n \geq 1$, entonces el término $F(x_1, \ldots, x_n)$ es Δ_n pues

$$\begin{aligned} x = F(x_1, \ldots, x_n) &\leftrightarrow (x_1 \in A_1 \wedge \cdots \wedge x_n \in A_n \wedge \phi(x_1, \ldots, x_n)) \\ &\quad \vee ((x_1 \notin A_1 \vee \cdots x_n \notin A_n) \wedge x = \varnothing) \\ &\leftrightarrow (x_1 \in A_1 \wedge \cdots \wedge x_n \in A_n \wedge \textstyle\bigwedge y(\phi(x_1, \ldots, x_n, y) \to y = x)) \\ &\quad \vee ((x_1 \notin A_1 \vee \cdots x_n \notin A_n) \wedge x = \varnothing). \end{aligned}$$

Es claro que al sustituir términos Δ_n en fórmulas Σ_n o Π_n la fórmula resultante sigue siendo Σ_n o Π_n (por el mismo argumento empleado tras la definición 4.22).

La tabla de la página siguiente muestra que los conceptos conjuntistas básicos son Δ_0.

5.2 Interpretaciones

Dedicamos esta sección a precisar la relación que se da entre teorías como B e IΣ_1. La idea básica la hemos señalado ya: las fórmulas de $\mathcal{L}_{\mathrm{tc}}$ se pueden ver como fórmulas de $\mathcal{L}_a$ sin más que entender que $x \in y$ no es la fórmula de $\mathcal{L}_{\mathrm{tc}}$ dada por el relator de pertenencia, sino la relación de pertenencia que hemos definido en ARP, o en cualquiera de las teorías aritméticas que hemos estudiado. En esta sección especificaremos qué hay que entender exactamente por "se pueden ver".

Conceptos $\boldsymbol{\Delta_0}$ en B

1. $z = x \cup y \leftrightarrow \bigwedge u \in z\,(u \in x \vee u \in y) \wedge \bigwedge u \in x\; u \in z \wedge \bigwedge u \in y\, u \in z$,
2. $z = x \cap y \leftrightarrow \bigwedge u \in z\,(u \in x \wedge u \in y) \wedge \bigwedge u \in x(u \in y \to u \in z)$,
3. $z = x \setminus y \leftrightarrow \bigwedge u \in z\,(u \in x \wedge u \notin y) \wedge \bigwedge u \in x(u \notin y \to u \in z)$,
4. $z = \varnothing \leftrightarrow \bigwedge u \in z\; u \neq u$,
5. $z = \bigcup x \leftrightarrow \bigwedge u \in z \bigvee v \in x\; u \in v \wedge \bigwedge v \in x \bigwedge u \in v\; u \in z$,
6. $x \subset y \leftrightarrow \bigwedge u \in x\; u \in y$,
7. $w = \{u, v\} \leftrightarrow v \in w \wedge v \in w \wedge \bigwedge x \in w\,(x = u \vee x = v)$,
8. $w = (u, v) \leftrightarrow \bigvee rs \in w(r = \{u\} \wedge s = \{u, v\})$
   $\wedge \bigwedge x \in w\; (x = \{u\} \vee x = \{u, v\})$,
9. $y = x' \leftrightarrow \bigwedge u \in y\; (u \in x \vee u = x) \wedge \bigwedge u \in x\; u \in y \wedge x \in y$,
10. x es transitivo $\leftrightarrow \bigwedge u \in x\; u \subset x$,
11. x es $\in$-conexo $\leftrightarrow \bigwedge uv \in x(u \in v \vee v \in u \vee u = v)$,
12. r es una relación $\leftrightarrow \bigwedge z \in r \bigvee w \in z \bigvee uv \in w\; z = (u, v)$,
13. r es una relación en $a \leftrightarrow \bigwedge z \in r \bigvee uv \in a\; z = (u, v)$,
14. f es una función $\leftrightarrow f$ es una relación $\wedge \bigwedge xy \in f \bigwedge r \in x \bigwedge s \in y$
    $\bigwedge uv \in r \bigwedge w \in s(x = (u, v) \wedge y = (u, w) \to v = w)$,
15. $f : x \longrightarrow y \leftrightarrow f$ es una función $\wedge \bigwedge z \in f \bigvee u \in x \bigvee v \in y\; z = (u, v)$
    $\wedge \bigwedge u \in x \bigvee v \in y \bigvee z \in f\; z = (u, v)$.

Definición 5.19 Sea S una teoría axiomática sobre un lenguaje formal de primer orden $\mathcal{L}_S$ y T otra teoría sobre otro lenguaje formal $\mathcal{L}_T$. Una *interpretación* M de S en T viene determinada por:

1. Un criterio[5] que a cada variable x de $\mathcal{L}_S$ le hace corresponder una variable $\bar{x}$ de $\mathcal{L}_T$ del mismo tipo (libre o ligada), de modo que variables distintas de $\mathcal{L}_S$ se correspondan con variables distintas de $\mathcal{L}_T$,
2. Una fórmula[6] $x \in M$ de $\mathcal{L}_T$ con x como única variable libre (que diremos que determina el *universo* de la interpretación) tal que $\underset{T}{\vdash} v|(v = v) \in M$,

[5]Si consideramos esta exposición formalizada en ARP, "criterio" debe entenderse como "funtor".

[6]La representamos $x \in M$ por convenio, igual que podríamos escribir $\phi(x)$, pero sin que esto presuponga que en $\mathcal{L}$ hay un relator $\in$. Como de costumbre, escribiremos $\bigwedge x \in M$, $\bigvee x \in M$ o $\overset{1}{\bigvee} x \in M$ con el significado obvio.

3. Para cada constante c de $\mathcal{L}_S$, un designador $\bar{c}$ de $\mathcal{L}_T$, tal que $\underset{T}{\vdash} \bar{c} \in M$,

4. Para cada funtor n-ádico f de $\mathcal{L}_S$, un término $\bar{f}(x_1, \ldots, x_n)$ de $\mathcal{L}_T$ con a lo sumo las variables libres indicadas tal que

$$\underset{T}{\vdash} \bigwedge x_1 \cdots x_n \in M \ \bar{f}(x_1, \ldots, x_n) \in M,$$

5. Para cada relator n-ádico R de $\mathcal{L}_S$ una fórmula $\bar{R}(x_1, \ldots, x_n)$ de $\mathcal{L}_T$ con a lo sumo las variables libres indicadas, de modo que la fórmula asociada al igualador de $\mathcal{L}_S$ es $x = y$,

de modo que, para todo axioma $\alpha(x_1, \ldots, x_n)$ de S con las variables libres entre las indicadas, se cumple que $\underset{T}{\vdash} \bigwedge \bar{u}_1 \cdots \bar{u}_n \in M\, \bar{\alpha}(\bar{u}_1, \ldots, \bar{u}_n)$, donde la *traducción* $\bar{\theta}$ a T de una semiexpresión θ de $\mathcal{L}_S$ se define como sigue:

1. $\bar{x}$ es la variable asociada a x por la interpretación,
2. $\bar{c}$ es el designador asociado a c por la interpretación,
3. $\overline{f(t_1, \ldots, t_n)} \equiv \bar{f}(\bar{t}_1, \ldots, \bar{t}_n)$,
4. $\overline{R(t_1, \cdots, t_n)} \equiv \bar{R}(\bar{t}_1, \ldots, \bar{t}_n)$,
5. $\overline{\neg\alpha} \equiv \neg\bar{\alpha}$,
6. $\overline{\alpha \vee \beta} \equiv \bar{\alpha} \vee \bar{\beta}$,
7. $\overline{\bigwedge u\alpha} \equiv \bigwedge \bar{u} \in M \ \bar{\alpha}$,
8. $\overline{\bigvee u\alpha} \equiv \bigvee \bar{u} \in M \ \bar{\alpha}$,
9. $\overline{u|\alpha} \equiv \bar{u}|(\bar{u} \in M \wedge \bar{\alpha})$.

En definitiva, $\bar{\theta}$ se obtiene sustituyendo cada signo de $\mathcal{L}_S$ por su interpretación en $\mathcal{L}_T$ y acotando todas las variables ligadas por el universo de la interpretación M.

Es inmediato que

$$\overline{\alpha \to \beta} \equiv \bar{\alpha} \vee \bar{\beta}, \quad \overline{\alpha \wedge \beta} \equiv \bar{\alpha} \wedge \bar{\beta}, \quad \overline{\alpha \leftrightarrow \beta} \equiv \bar{\alpha} \leftrightarrow \bar{\beta}$$

(entendiendo que los únicos conectores lógicos son el negador y el implicador, pero en caso contrario hay que añadir estas equivalencias a la definición de traducción, con lo que son igualmente válidas). También es fácil ver que la traducción de $\overset{1}{\bigvee} x\alpha$ es lógicamente equivalente a $\overset{1}{\bigvee} \bar{x} \in M\, \bar{\alpha}$.

En estos términos:

Teorema 5.20 $\mathrm{I}\Sigma_1$ *interpreta la teoría básica de conjuntos* B.

DEMOSTRACIÓN: Basta tomar como universo $x \in M \equiv x = x$ y como interpretación del único relator de $\mathcal{L}_{\rm tc}$ la fórmula $x \in y$ que hemos definido en ARP y que, por consiguiente, es definible también en ARP o en $\mathrm{I}\Sigma_1$.

La definición que hemos dado de interpretación está pensada para tratar con casos más generales, pero es claro que, para cualquier interpretación en la que $x \in M \equiv x = x$, cada fórmula $\bar\alpha$ es lógicamente equivalente a la que obtenemos si eliminamos de la definición de traducción toda referencia a M, de modo que traducimos $\bigwedge u\,\alpha$ por $\bigwedge \bar u\,\bar\alpha$, etc. Por lo tanto, la traducción de

$$\bigwedge u(u \in x \leftrightarrow u \in y) \to x = y$$

es (lógicamente equivalente a)

$$\bigwedge u(u \in x \leftrightarrow u \in y) \to x = y,$$

donde en la primera fórmula $\in$ es el relator de $\mathcal{L}_{\rm tc}$ y en la segunda $u \in x$ es la pertenencia definida en $\mathrm{I}\Sigma_1$.

Teniendo en cuenta las observaciones anteriores y posteriores al teorema 2.16, es inmediato que las traducciones de los axiomas de B son teoremas de $\mathrm{I}\Sigma_1$ (pues dichas traducciones son a su vez "traducciones" a $\mathcal{L}_a$ de los teoremas de ARP a los que estamos haciendo referencia), lo que demuestra que realmente tenemos una interpretación de B en $\mathrm{I}\Sigma_1$. ∎

Teorema 5.21 *Fijada una interpretación de una teoría S en una teoría T, para todo término $t(x_1,\ldots,x_n)$ de $\mathcal{L}_S$ se cumple*

$$\vdash_T \bigwedge \bar u_1 \cdots \bar u_n \in M\ \bar t(\bar u_1,\ldots,\bar u_n) \in M.$$

DEMOSTRACIÓN: Por inducción sobre la longitud de t. Si t es una variable es trivial. Si es una constante se cumple por definición de interpretación, si vale para $t_1,\ldots,t_m$ y $t \equiv f(t_1,\cdots,t_m)$, entonces

$$\vdash_T \bigwedge \bar u_1 \cdots \bar u_n \in M\ \bar t_i(\bar u_1,\ldots,\bar u_n) \in M$$

y por definición de interpretación $\vdash_T \bigwedge u_1 \cdots u_m \in M\ \bar f(u_1,\ldots,u_m) \in M$. De aquí se sigue fácilmente que $\vdash_T \bigwedge \bar u_1 \cdots \bar u_n \in M\ \bar f(\bar t_1,\ldots,\bar t_m) \in M$, pero esto es lo mismo que $\vdash_T \bigwedge \bar x_1 \cdots \bar x_n \in M\ \bar t(\bar x_1,\ldots,\bar x_n) \in M$.

Si $t \equiv u|\alpha$, entonces $\bar t = \bar u|(\bar u \in M \wedge \bar\alpha)$. Si es una descripción propia, entonces cumple trivialmente $\bar t \in M$, y si no lo es se cumple también, porque entonces la regla de las descripciones impropias nos da que $\bar t = v|v = v$ y estamos suponiendo que en T se demuestra $(v|v = v) \in M$. ∎

Teorema 5.22 *Fijada una interpretación de una teoría S en una teoría T, para toda variable x, todo término t y toda semiexpresión θ de $\mathcal{L}_S$ se cumple*

$$\overline{\mathrm{S}^t_x\theta} \equiv \mathrm{S}^{\bar t}_{\bar x}\bar\theta.$$

DEMOSTRACIÓN: Por inducción sobre la longitud de θ. Consideramos únicamente los casos menos obvios.

Si $\theta \equiv y$ es una variable, en el caso en que $x \not\equiv y$, tenemos que

$$\overline{\mathtt{S}^t_x\theta} \equiv \bar{y} \equiv \mathtt{S}^{\bar{t}}_{\bar{x}}\bar{y},$$

mientras que si $x \equiv y$, entonces $\overline{\mathtt{S}^t_x\theta} \equiv \bar{t} \equiv \mathtt{S}^{\bar{t}}_{\bar{x}}\bar{y}$.

Si $\theta \equiv \bigwedge u\,\alpha$ y $x \equiv u$ o bien x no está libre en α, tenemos que

$$\overline{\mathtt{S}^t_x\theta} \equiv \overline{\bigwedge u\,\alpha} \equiv \bigwedge \bar{u}(\bar{u} \in M \to \bar{\alpha}) \equiv \mathtt{S}^{\bar{t}}_{\bar{x}}\bigwedge \bar{u}(\bar{u} \in M \to \bar{\alpha}) \equiv \mathtt{S}^{\bar{t}}_{\bar{x}}\bar{\theta}.$$

En caso contrario, usando la hipótesis de inducción,

$$\overline{\mathtt{S}^t_x\theta} \equiv \overline{\bigwedge u\,\mathtt{S}^t_x\alpha} \equiv \bigwedge \bar{u}(\bar{u} \in M \to \mathtt{S}^{\bar{t}}_{\bar{x}}\bar{\alpha}) \equiv \mathtt{S}^{\bar{t}}_{\bar{x}}\bigwedge \bar{u}(\bar{u} \in M \to \bar{\alpha}) \equiv \mathtt{S}^{\bar{t}}_{\bar{x}}\bar{\theta}.$$

Los casos del particularizador y el descriptor son análogos. ∎

Ahora ya podemos probar el resultado fundamental sobre interpretaciones:

Teorema 5.23 *Fijada una interpretación de una teoría S en una teoría T, para toda fórmula $\alpha(x_1, \ldots, x_n)$ de $\mathcal{L}_S$, si se cumple $\underset{S}{\vdash} \alpha$, entonces*

$$\underset{T}{\vdash} \bigwedge \bar{u}_1 \cdots \bar{u}_n \in M\ \bar{\alpha}(\bar{u}_1, \ldots, \bar{u}_n).$$

En particular, si α es una sentencia y $\underset{S}{\vdash} \alpha$, entonces $\underset{T}{\vdash} \bar{\alpha}$.

DEMOSTRACIÓN: Si $\gamma(x_1, \ldots, x_n)$ es una fórmula cualquiera de $\mathcal{L}_S$, llamaremos $\bar{\gamma}^c$ a $\bigwedge \bar{u}_1 \cdots \bar{u}_n \in M\, \bar{\gamma}(\bar{u}_1, \ldots, \bar{u}_n)$, donde las x_i son las variables libres en γ. Veamos en primer lugar que si γ es un axioma lógico, entonces $\underset{T}{\vdash} \bar{\gamma}^c$.

Por simplificar la notación vamos a suponer que las variables libres en γ son x, y, aunque todos los argumentos que veremos valen sin cambio alguno cualquiera que sea el número de variables libres (y se simplifican si no hay ninguna).

Si γ es uno de los axiomas sobre conectores, es inmediato que $\bar{\gamma}$ es un axioma del mismo tipo, luego $\underset{T}{\vdash} \bar{\gamma}$, luego $\underset{T}{\vdash}(\bar{u} \in M \wedge \bar{v} \in M \to \bar{\gamma})$, luego, introduciendo generalizadores, $\underset{T}{\vdash} \bar{\gamma}^c$.

Si $\gamma \equiv \bigwedge u\alpha \to \mathtt{S}^t_u\alpha$, entonces $\bar{\gamma} \equiv \bigwedge \bar{u} \in M\bar{\alpha} \to S^{\bar{t}}_{\bar{u}}\bar{\alpha}$, y podemos razonar como sigue (la línea 2 es válida por 5.21):

1)	$\bar{x} \in M \wedge \bar{y} \in M$	Hipótesis
2)	$\bar{t} \in M$	Consecuencia de 1)
3)	$\bigwedge \bar{u} \in M\, \bar{\alpha}$	Hipótesis
4)	$\bar{t} \in M \to \mathtt{S}^{\bar{t}}_{\bar{u}}\bar{\alpha}$	EG 3
5)	$\mathtt{S}^{\bar{t}}_{\bar{u}}\bar{\alpha}$	MP 2, 4
6)	$\bigwedge \bar{u} \in M\, \bar{\alpha} \to \mathtt{S}^{\bar{t}}_{\bar{u}}\bar{\alpha}$	
7)	$\bar{x} \in M \wedge \bar{y} \in M \to \bigwedge \bar{u} \in M\, \bar{\alpha} \to \mathtt{S}^{\bar{t}}_{\bar{u}}\bar{\alpha}$	
8)	$\bigwedge \bar{u}_1\bar{u}_2 \in M(\bigwedge \bar{u} \in M\, \bar{\alpha} \to \mathtt{S}^{\bar{t}}_{\bar{u}}\bar{\alpha})$	IG 7

Si $\gamma \equiv \bigwedge u(\alpha \to \beta) \to (\alpha \to \bigwedge u\beta)$, entonces

$$\bar{\gamma} \equiv \bigwedge \bar{u} \in M\,(\bar{\alpha} \to \bar{\beta}) \to (\bar{\alpha} \to \bigwedge \bar{u} \in M\,\bar{\beta}).$$

Razonamos como sigue:

1)	$\bigwedge \bar{u} \in M(\bar{\alpha} \to \bar{\beta})$	Hipótesis
2)	$\bar{\alpha}$	Hipótesis
3)	$\bar{z} \in M$	Hipótesis
4)	$\bar{z} \in M \to (\bar{\alpha} \to \mathtt{S}^{\bar{z}}_{\bar{u}}\bar{\beta})$	EG 1
5)	$\bar{\alpha} \to \mathtt{S}^{\bar{z}}_{\bar{u}}\bar{\beta}$	MP 3, 4
6)	$\mathtt{S}^{\bar{z}}_{\bar{u}}\bar{\beta}$	MP2, 5
7)	$\bar{z} \in M \to \mathtt{S}^{\bar{z}}_{\bar{u}}\bar{\beta}$	
8)	$\bigwedge \bar{u} \in M\ \bar{\beta}$	IG 7
9)	$\bar{\alpha} \to \bigwedge \bar{u} \in M\ \bar{\beta}$	
10)	$\bigwedge \bar{u} \in M(\bar{\alpha} \to \bar{\beta}) \to (\bar{\alpha} \to \bigwedge \bar{u} \in M\ \bar{\beta})$	

De aquí podemos pasar a $\bar{\gamma}^c$ de forma obvia.

Si $\gamma \equiv \bigwedge u(u = t \to \alpha) \leftrightarrow \mathtt{S}^t_u\alpha$, entonces

$$\bar{\gamma} \equiv \bigwedge \bar{u} \in M(\overline{u} = \bar{t} \to \bar{\alpha}) \leftrightarrow \mathtt{S}^{\bar{t}}_{\bar{u}}\bar{\alpha}.$$

1)	$\bar{x} \in M \wedge \bar{y} \in M$	Hipótesis
2)	$\bar{t} \in M$	Consecuencia de 1)
3)	$\bigwedge \bar{u} \in M(\bar{u} = \bar{t} \to \bar{\alpha})$	Hipótesis
4)	$\bar{t} \in M \to (\bar{t} = \bar{t} \to \mathtt{S}^{\bar{t}}_{\bar{u}}\bar{\alpha})$	EG 3
5)	$\mathtt{S}^{\bar{t}}_{\bar{u}}\bar{\alpha}$	MP 2, 4, I
6)	$\bigwedge \bar{u} \in M(\bar{u} = \bar{t} \to \bar{\alpha}) \to \mathtt{S}^{\bar{t}}_{\bar{u}}\bar{\alpha}$	
7)	$\mathtt{S}^{\bar{t}}_{\bar{u}}\bar{\alpha}$	Hipótesis
8)	$\bigwedge \bar{u}(\bar{u} = \bar{t} \to \bar{\alpha})$	II 7
9)	$\bar{z} = \bar{t} \to \mathtt{S}^{\bar{z}}_{\bar{u}}\bar{\alpha}$	EG 8
10)	$\bar{z} \in M \to (\bar{z} = \bar{t} \to \mathtt{S}^{\bar{z}}_{\bar{u}}\bar{\alpha})$	Consecuencia de 9)
11)	$\bigwedge \bar{u} \in M(\bar{u} = \bar{t} \to \bar{\alpha})$	IG 10
12)	$\mathtt{S}^{\bar{t}}_{\bar{u}}\bar{\alpha} \to \bigwedge \bar{u} \in M(\bar{u} = \bar{t} \to \bar{\alpha})$	
13)	$\bigwedge \bar{u} \in M(\bar{u} = \bar{t} \to \bar{\alpha}) \leftrightarrow \mathtt{S}^{\bar{t}}_{\bar{u}}\bar{\alpha}$	IB 6, 12
14)	$\bar{x} \in M \wedge \bar{y} \in M \to \bigwedge \bar{u} \in M\,(\bar{u} = \bar{t} \to \bar{\alpha}) \leftrightarrow \mathtt{S}^{\bar{t}}_{\bar{u}}\bar{\alpha}$	
15)	$\bigwedge \bar{u}_1\bar{u}_2 \in M(\bigwedge \bar{u} \in M(\bar{u} = \bar{t} \to \bar{\alpha}) \leftrightarrow \mathtt{S}^{\bar{t}}_{\bar{u}}\bar{\alpha})$	IG 14

Si $\gamma \equiv \overset{1}{\bigvee} u\alpha \to \mathtt{S}^{u|\alpha}_u\alpha$, entonces es fácil ver que $\bar{\gamma}$ es lógicamente equivalente a

$$\overset{1}{\bigvee} \bar{u}(\bar{u} \in M \wedge \bar{\alpha}) \to \mathtt{S}^{\bar{u}|(\bar{u}\in M\wedge\bar{\alpha})}_{\bar{u}}\bar{\alpha},$$

y esto es un axioma lógico.

Si $\gamma \equiv \neg\overset{1}{\bigvee} u\alpha \to (u|\alpha) = (v|v = v)$, entonces $\bar{\gamma}$ es lógicamente equivalente a

$$\neg\overset{1}{\bigvee} \bar{u}(\bar{u} \in M \wedge \bar{\alpha}) \to \bar{u}|(\bar{u} \in M \wedge \bar{\alpha}) = \bar{v}|(\bar{v} \in M \wedge \bar{v} = \bar{v}).$$

Basta observar que $\bar{v}|(\bar{v} \in M \wedge \bar{v} = \bar{v}) = \bar{v}|(\bar{v} = \bar{v})$ para ver que $\bar{\gamma}$ es equivalente a un axioma lógico. Aquí hay que distinguir dos casos, según si $\overset{1}{\bigvee}\bar{y}\ \bar{y} \in M$ o bien $\neg\overset{1}{\bigvee}\bar{y}\ \bar{y} \in M$. En el primer caso la igualdad se tiene por la unicidad, ya que por la definición de interpretación tenemos que $\bar{v}|\bar{v} = \bar{v} \in M$. En el segundo las descripciones son impropias y llegamos a la misma igualdad mediante la regla de las descripciones impropias.

Tenemos así que las clausuras de los axiomas lógicos de $\mathcal{L}_S$ son teoremas de T, y las clausuras de los axiomas propios de S también son teoremas de T por definición de interpretación. Por último supongamos que tenemos una demostración de γ en S, digamos $\alpha_1, \ldots, \alpha_n$, y veamos por inducción sobre i que $\underset{T}{\vdash} \bar{\alpha}_i^c$. En particular así tendremos $\underset{T}{\vdash} \bar{\gamma}^c$.

Lo hemos probado ya si α_i es un axioma lógico o un axioma propio de S. Si α_i se deduce por MP, entonces tenemos dos líneas precedentes de la forma $\alpha \to \beta$ y α, con lo que, por hipótesis de inducción, en T tenemos $(\bar{\alpha} \to \bar{\beta})^c$ y $\bar{\alpha}^c$.

Supongamos que $\alpha \equiv \alpha(x, y, z)$ y $\beta \equiv \beta(x, w)$ (es decir, que las variables libres son las indicadas) y razonamos así (donde abreviamos $c \equiv v|v = v$):

1)	$\bigwedge \bar{x}\bar{y}\bar{z}\bar{w} \in M\ (\bar{\alpha} \to \bar{\beta})$	Premisa
2)	$\bigwedge \bar{x}\bar{y}\bar{z} \in M\, \bar{\alpha}$	Premisa
3)	$\bar{x} \in M \wedge c \in M \wedge c \in M \wedge \bar{w} \in M \to (\bar{\alpha}(\bar{x}, c, c) \to \bar{\beta})$	EG 1
4)	$\bar{x} \in M \wedge c \in M \wedge c \in M \to \bar{\alpha}(\bar{x}, c, c)$	EG 2
5)	$\bar{x} \in M \wedge \bar{w} \in M$	Premisa
6)	$c \in M$	Teorema
7)	$\bar{x} \in M \wedge c \in M \wedge c \in M \wedge \bar{w} \in M$	IC 5, 6
8)	$\bar{\alpha}(\bar{x}, c, c) \to \bar{\beta}(\bar{x}, \bar{w})$	MP 3, 7
9)	$\bar{\alpha}(\bar{x}, c, c)$	MP 4, 7
10)	$\bar{\beta}$	MP 8, 9
11)	$\bar{x} \in M \wedge \bar{w} \in M \to \bar{\beta}$	
12)	$\bigwedge \bar{u}_1\bar{u}_2 \in M\, \bar{\beta}(\bar{u}_1, \bar{u}_2)$	IG 11

Si α_i se deduce por generalización, tenemos una fórmula α previa tal que $\alpha_i \equiv \bigwedge u\, \mathtt{S}_x^u\alpha$ y, por hipótesis de inducción, en T se demuestra $\bar{\alpha}^c$. Si las variables libres de α_i son $u, x_1, \ldots, x_n$, es claro que $\bar{\alpha}^c$ es equivalente a

$$\bigwedge \bar{u}_1 \cdots \bar{u}_n\bar{u} \in M\, \bar{\alpha},$$

que a su vez equivale a $(\bigwedge \bar{u} \in M\, \bar{\alpha})^c$. ∎

Una consecuencia inmediata es que, en las condiciones del teorema anterior, si T es consistente también lo es S, pues la demostración de una contradicción en S permite demostrar una contradicción en T.

Otra consecuencia práctica es que si una teoría S se interpreta en otra T, fórmulas equivalentes en S tienen traducciones equivalentes en T.

Ahora ya podemos asegurar que todos los teoremas que hemos demostrado en B pueden verse como teoremas de $\mathrm{I}\Sigma_1$, en el sentido de que sus traducciones lo son, y sus traducciones son las fórmulas que denotamos del mismo modo, sólo que interpretando de forma diferente el signo $\in$.

5.3 La teoría de Zermelo

El teorema 5.16 contiene lo máximo que podemos obtener sobre números naturales en la teoría básica de conjuntos B. Si queremos definir la aritmética de los números naturales necesitamos un axioma adicional:

Definición 5.24 Llamaremos *teoría de conjuntos (restringida) de Zermelo* (Z^*) a la teoría axiomática sobre el lenguaje $\mathcal{L}_{\mathrm{tc}}$ cuyos axiomas son:

Extensionalidad	$\bigwedge xy(\bigwedge u(u \in x \leftrightarrow u \in y) \to x = y)$
Par	$\bigwedge xy\bigvee z\bigwedge u(u \in z \leftrightarrow u = x \vee u = y)$
Unión	$\bigwedge x\bigvee y\bigwedge u(u \in y \leftrightarrow \bigvee v(u \in v \wedge v \in x))$
Especificación	$\bigwedge x\bigvee y\bigwedge u(u \in y \leftrightarrow u \in x \wedge \phi(u))$

donde $\phi(z)$ es cualquier fórmula, tal vez con más variables libres.

Así, el esquema de especificación afirma que toda fórmula especifica un subconjunto de cada conjunto x, a saber, el conjunto y formado por los elementos de x que cumplen $\phi(u)$. Nuevamente, el axioma de extensionalidad nos da que el conjunto y es único, por lo que podemos definir

$$\{u \in x \mid \phi(u)\} \equiv y|\bigwedge u(u \in y \leftrightarrow u \in x \wedge \phi(u)),$$

que siempre es un conjunto, sea cual sea la fórmula $\phi(u)$.

En particular, en Z^* podemos probar el axioma de la diferencia, que es el único axioma de la teoría básica B que falta en Z^*. En efecto, la diferencia de dos conjuntos x e y es el conjunto $\{u \in x \mid u \notin y\}$.

Por lo tanto, Z^* es una extensión de B y todos los teoremas de B son también teoremas de Z^*. En particular, en Z^* está definida la clase ω de los números naturales.

Notas El esquema de especificación puede enunciarse diciendo que toda subclase de un conjunto es un conjunto. En efecto, una clase es de la forma $A = \{u \mid \phi(u)\}$, para cierta fórmula ϕ, y la fórmula $A \subset x$ significa que

$$\bigwedge u \in A(u \in x) \qquad \text{o, equivalentemente,} \qquad \bigwedge u(\phi(u) \to u \in x).$$

Por lo tanto, A tiene los mismos elementos que el conjunto $\{u \in x \mid \phi(u)\}$, y esto es lo que significa precisamente que A sea un conjunto.

Otra consecuencia es que la clase V de todos los conjuntos no puede ser un conjunto. Si lo fuera, también lo sería

$$R = \{x \mid x \notin x\} = \{x \in V \mid x \notin x\},$$

y tendríamos la paradoja de Russell: $R \in R \leftrightarrow R \notin R$. ∎

La relación entre fórmulas y conjuntos que establece el esquema de especificación nos permite demostrar en Z^* el principio de inducción de AP:

Teorema 5.25 *Para toda fórmula $\phi(x)$ (tal vez con más variables libres), la fórmula siguiente es un teorema de Z^*:*

$$\phi(0) \wedge \bigwedge n \in \omega\, (\phi(n) \to \phi(n')) \to \bigwedge n \in \omega\, \phi(n).$$

Demostración: Supongamos que $\phi(0) \wedge \bigwedge n \in \omega\, (\phi(n) \to \phi(n'))$, pero que existe un $m \in \omega$ tal que $\neg\phi(m)$. Sea $x = \{i \in m' \mid \neg\phi(i)\}$. Se trata de un conjunto no vacío, luego por definición de ordinal existe un $i \in x$ tal que $i \cap x = \varnothing$. Entonces i es un número natural tal que $\neg\phi(i)$. Por lo tanto $i \neq 0$, luego existe un $n \in i$ tal que $i = n'$. Por hipótesis $\neg\phi(n)$, luego $n \in i \cap x$, contradicción. ∎

Definiremos la suma y el producto de números naturales como casos particulares del teorema siguiente:

Teorema 5.26 *Sea X una clase, sea $G : \omega \times X \longrightarrow X$ y sea $a \in X$. Entonces existe una función $F : \omega \longrightarrow X$ tal que*

$$F(0) = a \wedge \bigwedge n \in \omega\ F(n') = G(n, F(n)).$$

Demostración: Definimos

$$F = \{(n, x) \in \omega \times X \mid \bigvee s(s : n' \longrightarrow X \wedge s(n) = x \wedge s(0) = a \wedge$$

$$\bigwedge i < n\ s(i') = G(i, s(i)))\}.$$

Veamos que F cumple lo pedido. Para ello probamos por inducción que

$$\bigwedge n \in \omega \bigvee x \in X\ (n, x) \in F.$$

En efecto, para $n = 0$ basta tomar $s = \{(0, a)\}$ y es claro que cumple lo necesario para justificar que $(0, a) \in F$. Si $(n, x) \in F$, existe s según la definición de F, y podemos considerar $s^* = s \cup \{(n', G(n, x))\}$, y es fácil ver que cumple lo requerido para justificar que $(n', G(n, x)) \in F$.

Si $s, s^* : n' \longrightarrow X$ cumplen la definición de F, entonces $\bigwedge i \in n'\ s(i) = s^*(i)$. Esto se prueba trivialmente por inducción sobre i. En particular, ahora es claro que

$$\bigwedge n \in \omega \overset{1}{\bigvee} x \in X\ (n, x) \in F,$$

es decir, que $F : \omega \longrightarrow X$. Ya hemos probado que $F(0) = a$. Dado $n \in \omega$, si $F(n') = x$, tomamos $s : n'' \longrightarrow X$ según la definición de F. Es claro entonces que $s|_{n'} : n' \longrightarrow X$ también cumple la definición de F, luego $F(n) = s(n)$ y $F(n') = s(n')$. Así concluimos que $F(n') = s(n') = G(n, s(n)) = G(n, F(n))$. ∎

Nota Si $X = \{u \mid \phi(u)\}$, $G = \{(n,u,v) \mid \psi(n,u,v)\}$, un enunciado sin clases propias del teorema anterior sería el siguiente: Dadas fórmulas $\phi(u)$ y $\psi(n,u,v)$ (tal vez con más variables libres), existe una fórmula $\chi(n,x,a)$ (con las mismas variables libres adicionales que las dos fórmulas dadas) tal que si

$$\bigwedge n \in \omega \bigwedge u(\phi(u) \to \overset{1}{\bigvee} v(\phi(v) \wedge \psi(n,u,v))),$$

entonces $\bigwedge n \in \omega \overset{1}{\bigvee} x(\phi(x) \wedge \chi(n,x,a)) \wedge \chi(0,a,a) \wedge$

$$\bigwedge n \in \omega \bigvee v_1 v_2(\phi(v_1) \wedge \phi(v_2) \wedge \chi(n,v_1,a) \wedge \chi(n',v_2,a) \wedge \psi(n,v_1,v_2)).$$

Concretamente:

$$\chi(n,x,a) \equiv n \in \omega \wedge \phi(x) \wedge \bigvee s(s \text{ es una función} \wedge \mathcal{D}s = n' \wedge \bigwedge i \in n'\ \phi(s(i))$$

$$\wedge\ s(n) = x \wedge s(0) = a \wedge \bigwedge i \in n\ \psi(i, s(i), s(i'))).$$

■

Si aplicamos este teorema a la función sucesor $S : \omega \longrightarrow \omega$ (es decir, a la fórmula $\psi(n,x,y) \equiv y = x'$), para cada $m \in \omega$ obtenemos una $F_m : \omega \longrightarrow \omega$ que cumple

$$F_m(0) = m \wedge \bigwedge n \in \omega\ F_m(n') = F_m(n)'.$$

Definimos $m + n = F_m(n)$. En estos términos,[7]

$$\bigwedge m \in \omega\ m + 0 = m \wedge \bigwedge mn \in \omega\ m + n' = (m+n)'.$$

Similarmente, si partimos de la función $G_m : \omega \longrightarrow \omega$ definida mediante $G_m(n) = n + m$, el teorema anterior nos da una función $F_m : \omega \longrightarrow \omega$ tal que $F_m(0) = 0 \wedge \bigwedge n \in \omega\, F_m(n') = F_m(n) + m$. Si llamamos $m \cdot n = F_m(n)$, tenemos que

$$\bigwedge m \in \omega\ m \cdot 0 = 0 \wedge \bigwedge mn \in \omega\ m \cdot n' = (m \cdot n) + m.$$

En definitiva, hemos demostrado lo siguiente:

Teorema 5.27 *En* Z^* *se demuestran los axiomas de Peano:*

1. $0 \in \omega$,
2. $\bigwedge n \in \omega\ n' \in \omega$,
3. $\bigwedge n \in \omega\ n' \neq 0$,
4. $\bigwedge mn \in \omega(m' = n' \to m = n)$,
5. $\bigwedge mn \in \omega\ m + 0 = m$,
6. $\bigwedge mn \in \omega\ m + n' = (m+n)'$,

[7]Más precisamente, el teorema anterior define una fórmula $\chi(n,x,m)$ tal que

$$F_m = \{(n,x) \mid \chi(n,x,m)\},$$

y entonces $m + n \equiv x|\chi(n,x,m)$.

7. $\bigwedge m \in \omega\ m \cdot 0 = 0$,
8. $\bigwedge mn \in \omega\ m \cdot n' = m \cdot n + m$,
9. $\phi(0) \wedge \bigwedge n \in \omega(\phi(n) \to \phi(n')) \to \bigwedge n \in \omega\, \phi(n)$,

para toda fórmula $\phi(n)$, tal vez con más variables libres.

Como consecuencia:

Teorema 5.28 *La teoría de Zermelo* Z^* *interpreta la aritmética de Peano* AP.

Demostración: Basta tomar como universo de la interpretación la fórmula $x \in \omega$, interpretar $\bar{0} \equiv 0 (\equiv \varnothing)$ y los funtores sucesor, suma y producto mediante los términos x', $x + y$, $x \cdot y$. Es claro que se cumplen todos los requisitos de la definición de interpretación.

Del teorema anterior se sigue fácilmente por inducción que

$$\bigwedge n \in \omega\, n' \in \omega, \quad \bigwedge mn \in \omega\, m + n \in \omega, \quad \bigwedge m \in \omega\, m \cdot n \in \omega,$$

así como que las traducciones de los axiomas de AP son teoremas de Z^*. Por lo tanto, todos los teoremas de AP pueden verse como teoremas de Z^* referentes a números naturales. ∎

Recíprocamente, vamos a ver que Z^* es interpretable en la aritmética de Peano. Como ya sabemos que la teoría básica B lo es (teorema 5.20), sólo falta probar que la traducción del esquema de especificación es demostrable en AP.

El teorema 2.16 nos dice que, para toda fórmula α de $\mathcal{L}_{\mathrm{arp}}$, existe un funtor que cumple el teorema

$$x \in \{u < k \mid \alpha(u, x_1, \ldots, x_n)\} \leftrightarrow x < k \wedge \alpha(x, x_1, \ldots, x_n).$$

Esto implica que en ARP (luego en $\mathrm{I}\Sigma_1$) se puede probar que

$$\bigvee y \bigwedge u(u \in y \leftrightarrow u < k \wedge \alpha(u, x_1, \ldots, x_n)),$$

pero no para toda fórmula α, sino únicamente para fórmulas atómicas (que son las que se corresponden con fórmulas de $\mathcal{L}_{\mathrm{arp}}$), y a lo sumo lo podemos extender para fórmulas Δ_0, pues toda fórmula Δ_0 es equivalente en ARP a una fórmula atómica. En $\mathrm{I}\Sigma_n$ podemos probar algo mejor:

Teorema 5.29 *Si $\alpha(x, x_1, \ldots, x_n)$ es una fórmula de tipo Δ_n en $\mathrm{I}\Sigma_n$, entonces*

$$\underset{\mathrm{I}\Sigma_n}{\vdash} \bigvee y \bigwedge u(u \in y \leftrightarrow u < k \wedge \alpha(u, x_1, \ldots, x_n)).$$

Demostración: La fórmula es[8] de tipo $\mathrm{I}\Sigma_n$, luego podemos razonar por inducción sobre k. Para $k = 0$ es trivial (se cumple con $y = \varnothing$). Si la suponemos

[8]Notemos que α tiene que ser Σ_n para que la implicación $\to$ sea Σ_n y tiene que ser Π_n para que la implicación $\leftarrow$ también lo sea, por lo que necesitamos que sea Δ_n.

cierta para k y llamamos y_0 a un conjunto que cumpla lo requerido para k, basta definir

$$y = \begin{cases} y_0 \cup \{k\} & \text{si} \quad \alpha(k, x_1, \ldots, x_n), \\ y_0 & \text{si } \neg\alpha(k, x_1, \ldots, x_n), \end{cases}$$

y así y cumple lo requerido para $k+1$. ∎

Esto es la versión aritmética del principio de especificación, pero ahora necesitamos una observación:

Como las semifórmulas atómicas $x = y$, $x \in y$ de $\mathcal{L}_{\mathrm{tc}}$ se traducen a las fórmulas de $\mathcal{L}_a$ que denotamos igual, y ambas son Δ_1, es inmediato comprobar que si α es una fórmula Δ_0 de $\mathcal{L}_{\mathrm{tc}}$, su traducción $\bar{\alpha}$ a $\mathcal{L}_a$ es una fórmula Δ_1 en $\mathrm{I}\Sigma_1$. Esto implica a su vez que si α es Σ_n o Π_n en B (o en cualquier extensión de B interpretable en $\mathrm{I}\Sigma_1$), entonces $\bar{\alpha}$ es del mismo tipo en la jerarquía de Kleene, lo cual implica que lo mismo vale para fórmulas Δ_n. Es decir:

> *Si α es una fórmula de tipo Σ_n, Π_n o Δ_n ($n \geq 1$) en $\mathcal{L}_{\mathrm{tc}}$ respecto de* B *(o de cualquier extensión de* B *interpretable en $\mathrm{I}\Sigma_1$) su traducción a $\mathcal{L}_a$ es una fórmula del mismo tipo respecto de $\mathrm{I}\Sigma_1$ (pero, en general, las fórmulas Δ_0 tienen traducciones Δ_1).*

Por consiguiente:

Teorema 5.30 *La traducción del esquema de especificación para fórmulas de tipo Δ_n ($n \geq 1$) respecto de* B *(o de cualquier teoría que extienda a* B *interpretable en $\mathrm{I}\Sigma_1$) es demostrable en $\mathrm{I}\Sigma_n$. En particular, la aritmética de Peano interpreta la teoría de Zermelo Z^*.*

Demostración: Si $\alpha(x, x_1, \ldots, x_n)$ es una fórmula de tipo Δ_n en una teoría interpretable en $\mathrm{I}\Sigma_1$, su traducción α_a es de tipo Δ_n en $\mathrm{I}\Sigma_1$, luego en $\mathrm{I}\Sigma_n$, y aplicando el teorema 5.29 a la fórmula $u \in x \wedge \alpha(u, x_1, \ldots, x_n)$ con $k = x$, obtenemos

$$\underset{\mathrm{I}\Sigma_n}{\vdash} \bigvee y \bigwedge u(u \in y \leftrightarrow u \in x \wedge \alpha_a(u, x_1, \ldots, x_n)).$$

Si α es arbitraria, entonces α_c será de tipo Δ_n para un n suficientemente grande y la traducción del axioma de especificación será demostrable en el $\mathrm{I}\Sigma_n$ correspondiente, luego en particular en AP. ∎

En particular acabamos de ver que si, en lugar de añadir a B todo el axioma de especificación, añadimos únicamente su restricción a fórmulas Δ_1 obtenemos una teoría interpretable en $\mathrm{I}\Sigma_1$, y la cuestión es si esta versión restringida basta para definir la suma y el producto de números naturales. Sin embargo, la respuesta es negativa y necesitaremos algunos axiomas adicionales, lo que nos lleva a la teoría de Kripke Platek que estudiamos en la sección siguiente.

5.4 La teoría de Kripke-Platek

La *teoría de conjuntos de Kripke-Platek* (KP) es la teoría axiomática sobre el lenguaje $\mathcal{L}_{\mathrm{tc}}$ cuyos axiomas son los siguientes:

Extensionalidad	$\bigwedge xy(\bigwedge u(u \in x \leftrightarrow u \in y) \to x = y)$
Par	$\bigwedge xy\bigvee z(x \in z \wedge y \in z)$
Unión	$\bigwedge x\bigvee y\bigwedge u \in x\bigwedge v \in u\ v \in y$
Δ_0-especificación	$\bigwedge x\bigvee y\bigwedge u(u \in y \leftrightarrow (u \in x \wedge \phi(u)))$ $(*)$
Δ_0-recolección	$\bigwedge u\bigvee v\ \phi(u,v) \to \bigwedge a\bigvee b\bigwedge u \in a\bigvee v \in b\ \phi(u,v)$ $(*)$
Π_1-regularidad	$\bigvee u\,\phi(u) \to \bigvee u\,(\phi(u) \wedge \bigwedge v \in u\,\neg\phi(v))$ $(**)$

$(*)$ Para toda fórmula $\phi(z)$ (tal vez con más variables libres) de tipo Δ_0,
$(**)$ Para toda fórmula $\phi(z)$ (tal vez con más variables libres) de tipo Π_1.

Veamos el papel que desempeña en esta teoría cada uno de sus tres esquemas de axioma. Empezamos por el primero: El esquema de Δ_0-especificación es una versión restringida del esquema de especificación de Z^*. Como en esta teoría, el axioma de extensionalidad nos da que el conjunto cuya existencia postula es único, por lo que podemos considerar el conjunto

$$\{u \in x \mid \phi(u)\} \equiv y|\bigwedge u(u \in y \leftrightarrow (u \in x \wedge \phi(u)))$$

La única diferencia con Z^* es que ahora sólo tenemos garantizado que este término es una descripción propia cuando la fórmula ϕ es Δ_0. No obstante, esto basta para demostrar los axiomas del conjunto vacío y de la diferencia de la teoría B, pues, tomando un conjunto cualquiera x, el conjunto $\{u \in x \mid u \neq u\}$ es un conjunto vacío (donde tenemos en cuenta que la fórmula $u \neq u$ es Δ_0) y el conjunto $\{u \in y \mid u \notin x\}$ es la diferencia de los conjuntos x e y. Por lo tanto KP es una extensión de la teoría básica B. En particular tenemos definida la clase ω de los números naturales (pero, en principio, no la suma y el producto que hemos definido en Z^* y que luego veremos que también son definibles en KP).

Del esquema de Π_1-regularidad extraemos dos consecuencias principales. La primera resulta de aplicarlo a la fórmula $\phi(u) \equiv u \in x$, lo que nos da que

$$\bigwedge x(x \neq \varnothing \to \bigvee u(u \in x \wedge u \cap x = \varnothing).$$

Como esto vale para todo conjunto, la definición de ordinal 5.2 puede simplificarse en KP hasta

$$x \in \Omega \leftrightarrow \bigwedge u \in x\ u \subset x \wedge \bigwedge uv \in x(u \in v \vee v \in u \vee u = v).$$

Por lo tanto, $x \in \Omega$ es una fórmula Δ_0 en KP (mientras que en B no lo es). De aquí se sigue que la fórmula $x \in \omega$ también es Δ_0.

En segundo lugar, es inmediato que el esquema de Π_1-regularidad equivale al resultado siguiente:

Principio de $\mathbf{\Sigma_1^\epsilon}$-inducción *Si $\phi(u)$ es una fórmula de clase Σ_1 (con posibles parámetros), entonces*

$$\bigwedge x(\bigwedge u \in x\ \phi(u) \to \phi(x)) \to \bigwedge x\, \phi(x).$$

Así, para demostrar que todo conjunto tiene una propiedad Σ_1 basta demostrar que un conjunto x la tiene bajo la hipótesis de inducción de que todos sus elementos la tienen. Si aplicamos esto a la fórmula

$$\Phi(x) \equiv (x \in \Omega \wedge \phi(x)) \vee x \notin \Omega$$

obtenemos:

Principio de $\mathbf{\Sigma_1}$-inducción transfinita *Si $\phi(u)$ es una fórmula Σ_1 (con posibles parámetros), entonces*

$$\bigwedge \alpha \in \Omega(\bigwedge \delta < \alpha\, \phi(\delta) \to \phi(\alpha)) \to \bigwedge \alpha \in \Omega\, \phi(\alpha).$$

Lo mismo vale (con la misma prueba) si cambiamos Ω por ω, de modo que para probar que todo número natural tiene una propiedad Σ_1 basta probar que uno la tiene supuesto que la tienen todos los menores que él. Finalmente, de aquí obtenemos la Σ_1-inducción usual:

Principio de $\mathbf{\Sigma_1}$-inducción *Si $\phi(u)$ es una fórmula Σ_1 (con posibles parámetros), entonces*

$$\phi(0) \wedge \bigwedge n \in \omega(\phi(n) \to \phi(n')) \to \bigwedge n \in \omega\, \phi(n).$$

Demostración: Supongamos $\phi(0) \wedge \bigwedge n \in \omega(\phi(n) \to \phi(n'))$ pero que, a pesar de ello, $\bigvee n \in \omega \neg\phi(n)$. Como $n \in \omega \wedge \neg\phi(n)$ es Π_1, por el axioma de regularidad existe un $n \in \omega$ tal que $\neg\phi(n) \wedge \bigwedge u \in n\, \phi(n))$. No puede ser $n = 0$, porque $\phi(0)$, luego $n = m'$, para cierto $m < n$, pero entonces $\phi(m)$, y por hipótesis también $\phi(n)$, contradicción. ∎

Nota El mismo argumento empleado en 4.24 muestra que el principio de inducción anterior es válido también para fórmulas Π_1. ∎

Pasamos ahora al esquema de Δ_0-recolección. Conviene observar que es equivalente a la siguiente versión "local":

$$\bigwedge a(\bigwedge u \in a \bigvee v\, \phi(u,v) \to \bigvee b \bigwedge u \in a \bigvee v \in b\, \phi(u,v))$$

Es obvio que la versión "local" implica la "global". Para el recíproco, si suponemos $\bigwedge u \in a \bigvee v\, \phi(u,v)$, aplicamos la versión global a la fórmula Δ_0

$$\Phi(u,v) \equiv (u \in a \wedge \phi(u,v)) \vee (u \notin a \wedge v = u).$$

La consecuencia que más nos va a interesar es el análogo del teorema 4.26:

Teorema 5.31 *Si T es una extensión de* KP, *las clases de fórmulas Σ_1^T y Π_1^T son cerradas para $\bigwedge x \in y, \bigvee x \in y$.*

DEMOSTRACIÓN: Supongamos que ϕ_0 es equivalente a $\bigvee z\,\phi$, donde ϕ es Δ_0, entonces, usando el axioma de Δ_0-recolección:

$$\bigwedge x \in y\,\phi_0(x) \leftrightarrow \bigwedge x \in y \bigvee z\,\phi(x,z) \leftrightarrow \bigvee y' \bigwedge x \in y \bigvee z \in y'\,\phi(x,z),$$

y la última fórmula es Σ_1. La clausura de una fórmula Π_1 respecto de $\bigvee x \in y$ se obtiene aplicando la parte ya probada a su negación. ∎

Ahora vamos a ver que en KP se puede probar, entre otras cosas, la Δ_1-especificación y la Σ_1-recolección. Para ello demostraremos en primer lugar una versión más precisa del teorema anterior. Necesitamos la definición siguiente:

Definición 5.32 Llamaremos semifórmulas $\tilde{\Sigma}_1$ en $\mathcal{L}_{\mathrm{tc}}$ a las que cumplen los criterios siguientes:

1. Toda semifórmula Δ_0 es $\tilde{\Sigma}_1$.
2. Si α y β son $\tilde{\Sigma}_1$, también lo son

$$\alpha \wedge \beta, \quad \alpha \vee \beta, \quad \bigwedge u \in x\,\alpha, \quad \bigvee u \in x\,\alpha, \quad \bigvee u\,\alpha.$$

Las semifórmulas $\tilde{\Pi}_1$ son las que se construyen del mismo modo pero cambiando $\bigvee u\,\alpha$ por $\bigwedge u\,\alpha$.

El teorema anterior muestra que las fórmulas $\tilde{\Sigma}_1$ y $\tilde{\Pi}_1$ son Σ_1 y Π_1 en sentido amplio, respectivamente, en KP, pero vamos a dar una prueba alternativa que nos dará una fórmula explícita equivalente de tipo Σ_1 o Π_1.

Si ϕ es una fórmula $\tilde{\Sigma}_1$ y x es una variable que no esté en ϕ, llamamos $\phi^{(x)}$ a la fórmula Δ_0 que resulta de sustituir en ϕ cada cuantificador no acotado $\bigvee u$ por $\bigvee u \in x$.

El teorema siguiente se demuestra trivialmente por inducción sobre la longitud de la fórmula:

Teorema 5.33 *Si ϕ es una fórmula de clase $\tilde{\Sigma}_1$ y x, y son variables que no están en ϕ, entonces se cumple:*

$$\phi^{(x)} \wedge x \subset y \to \phi^{(y)}, \qquad \phi^{(x)} \to \phi.$$

La versión explícita de los teoremas 5.18 y 5.31 para fórmulas $\tilde{\Sigma}_1$ es el teorema siguiente, del que obtendremos muchas consecuencias:

Teorema 5.34 ($\tilde{\boldsymbol{\Sigma}}_{\mathbf{1}}$-reflexión) *Si ϕ es una fórmula de clase $\tilde{\Sigma}_1$ y x es una variable que no está en ϕ, la fórmula $\phi \leftrightarrow \bigvee u\,\phi^{(u)}$ es un teorema de* KP.

DEMOSTRACIÓN: Una implicación es inmediata por el teorema anterior. Probamos la contraria por inducción sobre la longitud de ϕ. Si ϕ es de clase Δ_0 entonces $\phi \equiv \phi^{(x)}$ y el resultado es trivial.

Si $\phi \equiv \psi_1 \wedge \psi_2$, entonces $\phi^{(x)} \equiv \psi_1^{(x)} \wedge \psi_2^{(x)}$. Por hipótesis de inducción tenemos que

$$\psi_1 \leftrightarrow \bigvee u\, \psi_1^{(u)}, \qquad \psi_2 \leftrightarrow \bigvee u\, \psi_2^{(u)}.$$

Si se cumple $\phi \equiv \psi_1 \wedge \psi_2$, entonces existen x_1 y x_2 tales que $\psi_1^{(x_1)}$ y $\psi_2^{(x_2)}$. Si llamamos $x = x_1 \cup x_2$ entonces tenemos $\psi_1^{(x)} \wedge \psi_2^{(x)}$ por el teorema anterior, luego $\bigvee u\, \phi^{(u)}$.

Si $\phi \equiv \psi_1 \vee \psi_2$ la equivalencia es trivial:

$$\phi \leftrightarrow \bigvee u\, \psi_1^{(u)} \vee \bigvee u\, \psi_2^{(u)} \leftrightarrow \bigvee u(\psi_1^{(u)} \vee \psi_2^{(u)}) \leftrightarrow \bigvee u\, \phi^{(u)}.$$

Si $\phi \equiv \bigvee v \in y\, \psi$, entonces

$$\phi \leftrightarrow \bigvee v \in y \bigvee u\, \psi^{(u)} \leftrightarrow \bigvee u \bigvee v \in y\, \psi^{(u)} \equiv \bigvee u\, \phi^{(u)}.$$

Si $\phi \equiv \bigwedge v \in y\, \psi$, entonces, por hipótesis de inducción, $\phi \leftrightarrow \bigwedge v \in y \bigvee u\, \psi^{(u)}$. Suponiendo ϕ, por Δ_0-recolección existe un w tal que $\bigwedge v \in y \bigvee u \in w\, \psi^{(u)}$. Sea $w' = \bigcup w$. Así $\bigwedge v \in y\, \psi^{(w')}$, luego $\bigvee u \bigwedge v \in y\, \psi^{(u)} \equiv \bigvee u\, \phi^{(u)}$.

Si $\phi \equiv \bigvee v\, \psi$ y suponemos ϕ, entonces sea x tal que $\psi(x)$. Por hipótesis de inducción existe un w tal que $\psi^{(w)}(x)$. Sea $w' = w \cup \{x\}$. Entonces también $\psi^{(w')}$, luego $\bigvee v \in w'\, \psi^{(w')} \equiv \phi^{(w')}$, luego $\bigvee u\, \phi^{(u)}$. ∎

Ahora ya podemos "mejorar" los axiomas de KP:

Teorema 5.35 ($\boldsymbol{\Sigma_1}$-recolección) *Para toda fórmula ϕ de clase Σ_1 (tal vez con más variables libres), la fórmula siguiente es un teorema de* KP:

$$\bigwedge u \in x \bigvee v\, \phi(u,v) \to \bigvee y(\bigwedge u \in x \bigvee v \in y\, \phi(u,v) \wedge \bigwedge v \in y \bigvee u \in x\, \phi(u,v)).$$

Demostración: Basta probarlo para fórmulas de clase $\tilde{\Sigma}_1$, pues en particular éstas incluyen a todas las fórmulas Σ_1 en sentido estricto, y a su vez, si el teorema se cumple para ellas, se cumple también para todas las fórmulas Σ_1 en sentido amplio (las equivalentes a fórmulas Σ_1 en sentido estricto).

Por $\tilde{\Sigma}_1$-reflexión, supuesto $\bigwedge u \in x \bigvee v\, \phi(u,v)$, existe un z tal que

$$\bigwedge u \in x \bigvee v \in z\, \phi^{(z)}(u,v).$$

Por Δ_0-especificación existe el conjunto

$$y = \{v \in z \mid \bigvee u \in x\, \phi^{(z)}(u,v)\},$$

y claramente cumple lo pedido. ∎

Teorema 5.36 ($\boldsymbol{\Sigma_1}$-reducción) *Si ϕ es una fórmula Π_1 y ψ es Σ_1, la fórmula siguiente es un teorema de* KP:

$$\bigwedge u \in x(\phi(u) \to \psi(u)) \to$$

$$\bigvee y(\bigwedge u \in x(\phi(u) \to u \in y) \wedge \bigwedge u \in y(u \in x \wedge \psi(u))).$$

DEMOSTRACIÓN: No perdemos generalidad si suponemos que ϕ es Π_1 en sentido estricto y ψ es Σ_1 en sentido estricto. Pongamos, concretamente, que $\phi \equiv \bigwedge v\, \phi'$. Supongamos

$$\bigwedge u \in x(\phi(u) \to \psi(u))$$

o, lo que es lo mismo, $\bigwedge u \in x(\bigvee v \neg\phi'(u) \vee \psi(u))$. Esta fórmula es $\tilde{\Sigma}_1$, luego por $\tilde{\Sigma}_1$-reflexión existe un z tal que $\bigwedge u \in x(\phi^{(z)}(u) \to \psi^{(z)}(u))$. Basta tomar $y = \{u \in x \mid \psi^{(z)}(u)\}$, que existe por Δ_0-especificación. Así, si $u \in x$ cumple $\phi(u)$, pero $u \notin y$, entonces $\neg\psi^{(z)}(u)$, pero esto implica $\neg\phi^{(z)}(u)$, luego $\neg\phi(u)$, y tenemos una contradicción, luego $u \in y$. ∎

De aquí se sigue inmediatamente:

Teorema 5.37 ($\boldsymbol{\Delta_1}$-especificación) *Si ϕ es una fórmula Π_1 y ψ es Σ_1, la fórmula siguiente es un teorema de* KP:

$$\bigwedge u \in x(\phi(u) \leftrightarrow \psi(u)) \to \bigvee y \bigwedge u(u \in y \leftrightarrow u \in x \wedge \psi(u)).$$

Observemos que lo que afirma el teorema anterior es que las fórmulas Δ_1^T, para cualquier teoría T que extienda a KP, también definen subconjuntos de un conjunto dado. Así pues,

$$\{u \in x \mid \phi(u)\}$$

es una descripción propia siempre que ϕ es una fórmula Δ_1.

De aquí podemos deducir un último resultado general de interés, pero para ello necesitamos probar antes algunos hechos básicos, empezando por la existencia de productos cartesianos:

Teorema 5.38 $\overset{1}{\bigvee} z \bigwedge w(w \in z \leftrightarrow \bigvee u \in x \bigvee v \in y\ w = (u, v))$.

DEMOSTRACIÓN: La unicidad es consecuencia del axioma de extensionalidad. Basta probar la existencia. Aplicamos Δ_0-recolección a la fórmula $\phi(u, w) \equiv w = (u, v)$, y obtenemos un b tal que $\bigwedge u \in x\ (u, v) \in b$. Ahora consideramos la fórmula Δ_0

$$\phi(v, t) \equiv \bigwedge u \in x \bigvee w \in t\ w = (u, v).$$

Así, $\phi(v, t)$ afirma que t contiene todos los pares (u, v) con $u \in x$. Acabamos de probar que para todo v existe un tal t, luego podemos aplicar de nuevo el axioma de recolección, que nos da un b tal que

$$\bigwedge v \in y \bigvee t \in b \bigwedge u \in x\, (u, v) \in t.$$

Ahora tomamos $a = \bigcup b$, de modo que si $u \in x \wedge v \in y$, entonces existe un $t \in b$ tal que $(u, v) \in t$, luego $(u, v) \in a$. Basta tomar

$$z = \{w \in a \mid \bigvee u \in x \bigvee v \in y\ w = (u, v)\}.$$

∎

Por lo tanto:

$$x \times y \equiv z \mid \bigwedge w(w \in z \leftrightarrow \bigvee u \in x \bigvee v \in y\, w = (u,v))$$

es una descripción propia. Si $\phi(u,v)$ es una fórmula Δ_1, podemos considerar el conjunto

$$\{(u,v) \in x \times y \mid \phi(u,v)\} \equiv \{w \in x \times y \mid \bigvee u \in x \bigvee v \in y\ (w = (u,v) \wedge \phi(u,v))\},$$

pues la fórmula que lo especifica es también Δ_1.

A partir de aquí ya es fácil probar que todos los conceptos conjuntistas básicos definen conjuntos. Conviene observar en general que

$$\bigwedge x \in \bigcup y\, \alpha \leftrightarrow \bigwedge u \in y \bigwedge x \in u\ \alpha, \qquad \bigvee x \in \bigcup y\, \alpha \leftrightarrow \bigvee u \in y \bigvee x \in u\ \alpha,$$

por lo que estas acotaciones de cuantificadores no aumentan el tipo de una fórmula. Esto es útil porque $u, v \in \{u,v\} \in (u,v)$, luego $u, v \in \bigcup(u,v)$. Teniendo esto en cuenta vemos que podemos definir como sigue el dominio y el rango de un conjunto:

$$\mathcal{D}f = \{u \in \bigcup\bigcup f \mid \bigvee w \in f \bigvee v \in \bigcup w\ w = (u,v)\},$$

$$\mathcal{R}f = \{v \in \bigcup\bigcup f \mid \bigvee w \in f \bigvee u \in \bigcup w\ w = (u,v)\},$$

de modo que

$$x \in \mathcal{D}f \leftrightarrow \bigvee y\, (x,y) \in f, \qquad y \in \mathcal{D}f \leftrightarrow \bigvee x\, (x,y) \in f.$$

Con esto ya es fácil definir por Δ_1-especificación todos los conceptos relacionados con aplicaciones que introdujimos en 2.19 para ARP, empezando por el propio concepto de aplicación:

$$f : x \longrightarrow y \equiv f \subset x \times y \wedge \bigwedge u \in f \overset{1}{\bigvee} v \in y\, (u,v) \in f.$$

Similarmente definimos los conceptos de aplicación inyectiva, suprayectiva, biyectiva, la composición de aplicaciones, etc. Por ejemplo:

$$f \circ g = \{(u,w) \in \mathcal{D}f \times \mathcal{R}g \mid \bigvee v \in \bigcup\bigcup f\ (u,v) \in f \wedge (v,w) \in g\},$$

$$f^{-1} = \{(v,u) \in \mathcal{R}f \times \mathcal{D}f \mid (u,v) \in f\}.$$

Además, es pura rutina probar que los conceptos conjuntistas básicos son Δ_0. La tabla de la página siguiente muestra los más importantes. No hemos incluido algunos como "$f : x \longrightarrow y$" o "$f : x \longrightarrow y$ inyectiva", etc., porque la comprobación es inmediata. Por otro lado, ya hemos señalado que, en virtud del axioma de Π_1-regularidad, la fórmula $x \in \omega$ es Δ_0.

Ahora ya podemos probar:

Teorema 5.39 (Σ_1-reemplazo) *Para toda fórmula ϕ de tipo Σ_1 se cumple:*

$$\bigwedge u \in x \overset{1}{\bigvee} v\, \phi(u,v) \to \bigvee f y(f : x \longrightarrow y \text{ suprayectiva} \wedge \bigwedge u \in x\, \phi(u, f(u))).$$

Conceptos $\boldsymbol{\Delta_0}$ en KP

1. $z = x \times y \leftrightarrow \bigwedge u \in x \bigwedge v \in y \bigvee w \in z\ w = (u,v)$
   $\wedge \bigwedge w \in z \bigvee u \in x \bigvee v \in y\ w = (u,v)$,
2. $z = \mathcal{D}x \leftrightarrow \bigwedge w \in z \bigvee u \in x \bigvee r \in w \bigvee v \in r\ w = (u,v)$
   $\wedge \bigwedge u \in x \bigvee w \in z \bigvee r \in w \bigvee v \in r\ w = (u,v)$,
3. $z = \mathcal{R}x \leftrightarrow \bigwedge w \in z \bigvee v \in x \bigvee r \in w \bigvee u \in r\ w = (u,v)$
   $\wedge \bigwedge v \in x \bigvee w \in z \bigvee r \in w \bigvee u \in r\ w = (u,v)$,
4. $y = f[x] \leftrightarrow \bigwedge v \in y \bigvee u \in x \bigvee z \in f\ z = (u,v)$
   $\wedge \bigwedge z \in f \bigwedge u \in x \bigwedge w \in z \bigwedge v \in w (z = (u,v) \to v \in y)$,
5. $x = f^{-1}[y] \leftrightarrow \bigwedge u \in x \bigvee v \in y \bigvee z \in f\ z = (u,v))$
   $\wedge \bigwedge v \in y \bigwedge z \in f \bigwedge w \in z \bigwedge u \in w (z = (u,v) \to u \in x)$,
6. $y = x^{-1} \leftrightarrow \bigwedge z \in y \bigvee w \in z \bigvee uv \in w \bigvee z' \in x (z = (u,v) \wedge z' = (v,u))$
   $\wedge \bigwedge z \in x \bigwedge w \in z \bigwedge uv \in w (z = (u,v) \to \bigvee z' \in y\ z' = (v,u))$,
7. $g = f|_x \leftrightarrow g \subset f \wedge \bigwedge z \in f \bigwedge w \in z \bigwedge uv \in w (z = (u,v) \wedge u \in x \to z \in g)$
   $\wedge \bigwedge z \in g \bigvee w \in z \bigvee u \in x \bigvee v \in w\ z = (u,v)$,
8. $h = f \circ g \leftrightarrow \bigwedge z \in h \bigvee z' \in f \bigvee z'' \in g \bigvee w' \in z' \bigvee w'' \in z'' \bigvee uu' \in w'$
   $\bigvee u'' \in w''(z = (u,u'') \wedge z' = (u,u') \wedge z'' = (u',u''))$
   $\wedge \bigwedge z' \in f \bigwedge z'' \in g \bigvee z \in h \bigvee w' \in z' \bigvee w'' \in z'' \bigvee uu' \in w' \bigvee u'' \in w''$
   $(z = (u,u'') \wedge z' = (u,u') \wedge z'' = (u',u''))$.

Demostración: Supongamos que $\bigwedge u \in x \overset{1}{\bigvee} v\, \phi(u,v)$. Por Σ_1-recolección, existe un z tal que $\bigwedge u \in x \bigvee v \in z\, \phi(u,v)$ y, claramente, de hecho,

$$\bigwedge u \in x \overset{1}{\bigvee} v \in z\, \phi(u,v).$$

Ahora observamos que, para todo $w \in x \times z$ es equivalente

$$\bigvee uv(w = (u,v) \wedge \phi(u,v)) \leftrightarrow \neg \bigvee uvv'(w = (u,v) \wedge v \neq v' \wedge \phi(u,v')),$$

y ambas fórmulas son Σ_1 y Π_1 respectivamente, luego por Δ_1-especificación existe el conjunto

$$f = \{(u,v) \in x \times z \mid \phi(u,v)\},$$

y cumple lo pedido con $y = \mathcal{R}x$. ∎

Terminamos esta sección probando que con KP "vamos por el buen camino":

Teorema 5.40 KP *es interpretable en* $\mathrm{I}\Sigma_1$.

DEMOSTRACIÓN: Por el teorema 5.30 sabemos que la traducción de una instancia de Δ_0-especificación (o incluso de Δ_1-especificación) es demostrable en $\mathrm{I}\Sigma_1$.

La traducción de una instancia del axioma de Δ_0-recolección es una fórmula del mismo aspecto:

$$\bigwedge u \bigvee v\ \bar{\phi}(u,v) \to \bigwedge a \bigvee b \bigwedge u \in a \bigvee v \in b\ \bar{\phi}(u,v),$$

sólo que la traducción $\bar{\phi}$ es una fórmula de tipo Δ_1 en $\mathcal{L}_a$. Vamos a ver que esto es un teorema de $\mathrm{I}\Sigma_1$ incluso para fórmulas de tipo Σ_1 (lo cual no es sorprendente, pues hemos visto que la Σ_1-especificación es demostrable en KP).

Para ello usamos el teorema de Σ_1-recolección demostrado en 4.27:

$$\bigwedge u \le x \bigvee v\, \phi(u,v) \to \bigvee w \bigwedge u \le x \bigvee v \le w\, \phi(u,v).$$

Concretamente, lo aplicamos a la fórmula Σ_1:

$$\psi(u,v) \equiv (u \in x \wedge \bar{\phi}(u,v)) \vee (u \notin x \wedge v = 0),$$

que, suponiendo $\bigwedge u \bigvee v\, \bar{\phi}(u,v)$, cumple $\bigwedge u \le x \bigvee v\, \psi(u,v)$, luego nos da la existencia de un conjunto w tal que

$$\bigwedge u \le x \bigvee v \le w((u \in x \wedge \bar{\phi}(u,v)) \vee (u \notin x \wedge v = 0)).$$

En particular,

$$\bigwedge u \in x \bigvee v \le w\, \bar{\phi}(u,v).$$

Basta tomar $y = I_{w+1}$ (el conjunto de los números menores que $w+1$), y así

$$\bigwedge u \in x \bigvee v \in y\, \bar{\phi}(u,v).$$

La traducción del axioma de Π_1-regularidad se cumple tomando el mínimo u que cumple $\phi(u)$, que existe por el teorema 4.29. ∎

Por lo tanto, todo teorema de KP puede verse como un teorema de $\mathrm{I}\Sigma_1$.

5.5 KP como teoría aritmética

Veamos ahora que la demostración (en Z^*) del teorema 5.26 es válida también en KP. En primer lugar, por el teorema 5.31, la fórmula $\chi(n,x,a)$ mostrada explícitamente en la nota posterior es Σ_1 si lo son las fórmulas ϕ y ψ, y en segundo lugar observamos que todas las inducciones que se realizan en la prueba son respecto de fórmulas Σ_1. La conclusión es que en KP se demuestra el siguiente teorema de recursión):

Teorema 5.41 *Sea X una clase, sea $G : \omega \times X \longrightarrow X$ y sea $a \in X$. Si X y G son de tipo Σ_1, existe una función $F : \omega \longrightarrow X$ de tipo Σ_1 tal que*

$$F(0) = a \wedge \bigwedge n \in \omega\ F(n') = G(n, F(n)).$$

Esto basta para definir en KP la suma y el producto de números naturales, pues para la suma podemos aplicar el teorema anterior a la clase $X = \omega$ (que es de tipo Δ_0) y a la función $G(i, n) = m \leftrightarrow m = n'$ (también Δ_0). Esto nos da el término $m + n$ de tipo Σ_1 y, por consiguiente, Δ_1.

A su vez, para definir el producto aplicamos el teorema de nuevo a $X = \omega$ y a la función $G(i, n) = r \leftrightarrow r = n + m$, que es Σ_1, luego el producto $m \cdot n$ es un término Δ_1. Ahora podemos probar:

Teorema 5.42 (Axiomas de Peano) *Existen términos x', $x + y$, $x \cdot y$, 0 del lenguaje $\mathcal{L}_{\mathrm{tc}}$, todos ellos de tipo Δ_1, tales que en* KP *se demuestra:*

1. $0 \in \omega$
2. $\bigwedge n \in \omega\ n' \in \omega$,
3. $\bigwedge n \in \omega\, n' \neq 0$,
4. $\bigwedge mn \in \omega(m' = n' \to m = n)$,
5. $\bigwedge m \in \omega\ m + 0 = m$,
6. $\bigwedge mn \in \omega\ m + n' = (m + n)'$,
7. $\bigwedge m \in \omega\ m \cdot 0 = 0$,
8. $\bigwedge mn \in \omega\ m \cdot n' = m \cdot n + m$,
9. $\phi(0) \wedge \bigwedge n \in \omega(\phi(n) \to \phi(n')) \to \bigwedge n \in \omega\, \phi(n)$, *para toda fórmula $\phi(x)$ de tipo Σ_1, tal vez con más variables libres.*

Este teorema casi afirma que KP interpreta a $\mathrm{I}\Sigma_1$. En efecto, todas las condiciones se cumplen trivialmente, salvo que las traducciones de los casos particulares del principio de inducción de $\mathrm{I}\Sigma_1$ sean teoremas de KP, pues hay que tener presente que la jerarquía de Lévy no es la misma que la jerarquía de Kleene, de modo que en principio Σ_1 significa algo distinto en $\mathrm{I}\Sigma_1$ y en KP. Lo que sí que es inmediato es que KP interpreta a la teoría Q. En particular, para cada expresión θ de $\mathcal{L}_a$ podemos considerar su traducción[9] θ_{tc} a $\mathcal{L}_{\mathrm{tc}}$ en el sentido de 5.19.

Ahora, si t es un término sin descriptores de $\mathcal{L}_a$, una simple inducción sobre la longitud de t prueba que t_{tc} es un término Δ_1 en KP (puesto que todos los conceptos aritméticos están definidos en KP mediante términos y fórmulas Δ_1).

Por lo tanto, si t_1 y t_2 son términos sin descriptores de $\mathcal{L}_a$, entonces

$$(t_1 = t_2)_{\mathrm{tc}} \equiv t_{1\mathrm{tc}} = t_{2\mathrm{tc}}, \qquad (t_1 \leq t_2)_{\mathrm{tc}} \equiv \bigvee z \in \omega\ z + t_{1\mathrm{tc}} = t_{2\mathrm{tc}}$$

son fórmulas Σ_1 en KP. De aquí se sigue inmediatamente que la traducción de toda fórmula abierta de $\mathcal{L}_a$ es una fórmula Σ_1 en KP, luego la traducción de un caso del principio de inducción correspondiente a una fórmula abierta

[9] Identificaremos las variables de $\mathcal{L}_a$ y $\mathcal{L}_{\mathrm{tc}}$ y siempre consideraremos que la traducción de una variable es ella misma.

es un caso de Σ_1-inducción en KP, luego es un teorema de KP. Esto prueba que KP interpreta a IA. En particular, tenemos que KP prueba que la suma es conmutativa, etc.

El problema que tenemos que abordar es que la fórmula de $\mathcal{L}_a$ que define la relación de orden:

$$x \leq y \equiv \bigvee u\, u + x = y$$

es Δ_0 en $\mathcal{L}_a$ por definición, pero se traduce a una fórmula

$$x \leq_{\mathrm{tc}} y \equiv \bigvee u \in \omega\, u + x = y$$

que en principio es Σ_1 en $\mathcal{L}_{\mathrm{tc}}$, pero, para que la jerarquía de Kleene se corresponda con la de Lévy, necesitamos que sea como máximo Δ_1. El teorema siguiente prueba que en realidad es Δ_0:

Teorema 5.43 *La fórmula $x \leq_{\mathrm{tc}} y$ que traduce la relación de orden en* $\mathrm{I}\Sigma_1$ *cumple (en* KP*)*

$$\bigwedge xy \in \omega(x \leq_{\mathrm{tc}} y \leftrightarrow x \leq y),$$

donde $x \leq y \equiv x \subset y$ es la relación de orden usual en ω. Por consiguiente,

$$\bigwedge xy \in \omega(x <_{\mathrm{tc}} y \leftrightarrow x \in y).$$

Demostración: Veamos por inducción sobre y que

$$x \in \omega \wedge y \in \omega \wedge x \leq y \to x \leq_{\mathrm{tc}} y.$$

Notemos que se trata de una fórmula Σ_1, por lo que la inducción es lícita en KP. Obviamente, si $x \leq_c 0$, tenemos que $x = 0$, luego $x = 0 + 0$ y $\bigvee u \in \omega\, u + x = 0$.

Si la implicación es válida para y y se cumple $x \leq y'$, el teorema 5.13 nos da que $x \leq y$ o $x = y'$ (pues si fuera $y < x$, entonces $y' \leq x$ y de hecho $x = y'$). En el primer caso, por hipótesis de inducción, existe un $z \in \omega$ tal que $x + z = y$, luego por la definición de suma, $x + z' = y'$. En el segundo caso $x + 0 = y'$, luego en ambos casos $\bigvee u \in \omega\ u + x = y'$.

Una inducción mucho más simple prueba que

$$x \in \omega \wedge z \in \omega \to x \leq x + z,$$

de donde

$$x \in \omega \wedge y \in \omega \wedge x \leq_{\mathrm{tc}} y \to x \leq y.$$

Por lo tanto, $\bigwedge xy \in \omega(x \leq_{\mathrm{tc}} y \leftrightarrow x \leq y)$. ∎

Ahora estamos en condiciones de probar lo siguiente:

Teorema 5.44 *Si γ es una fórmula Σ_1, Π_1 o Δ_1 de $\mathcal{L}_a$ con variables libres $x_1, \ldots, x_n$, entonces existe una fórmula γ^* del mismo tipo en* KP *(con las mismas variables libres) tal que en* KP *se demuestra*

$$x_1 \in \omega \wedge \cdots \wedge x_n \in \omega \to (\gamma_{\mathrm{tc}} \leftrightarrow \gamma^*).$$

DEMOSTRACIÓN: Veamos en primer lugar que si partimos de una fórmula γ de tipo Δ_0 obtenemos una fórmula γ^* de tipo Δ_1 que cumple el teorema.

En efecto, razonamos por inducción[10] sobre la longitud de γ. Si $\gamma \equiv t_1 = t_2$, entonces $\gamma_{\rm tc} \equiv t_{1\rm tc} = t_{2\rm tc}$ es Δ_1, luego basta tomar $\gamma^* \equiv \gamma_{\rm tc}$. Si $\gamma \equiv t_1 \le t_2$, entonces $\gamma_{\rm tc} \equiv t_{1\rm tc} \le_{\rm tc} t_{2\rm tc}$. Si $x_1, \ldots, x_n \in \omega$, sabemos que $t_{1\rm tc}, t_{2\rm tc} \in \omega$, luego

$$\gamma_{\rm tc} \leftrightarrow t_{1\rm tc} \subset t_{2\rm tc},$$

y la fórmula $t_{1\rm tc} \subset t_{2\rm tc}$ es Δ_1.

Si el teorema es cierto para γ_1 y γ_2, es inmediato que también vale para $\neg\gamma_1$ y $\gamma_1 \vee \gamma_2$.

Supongamos que $\gamma \equiv \bigwedge u \le y\, \alpha(u)$, de modo que, por hipótesis de inducción, existe $\alpha^*(x)$ de tipo Δ_1 tal que en KP se demuestra

$$x \in \omega \wedge x_1 \in \omega \wedge \cdots x_n \in \omega \to (\alpha_{\rm tc}(x) \leftrightarrow \alpha^*(x)).$$

Entonces en KP se demuestra también que $\gamma_{\rm tc}$ equivale a

$$\bigwedge u(u \in \omega \wedge u \le_{\rm tc} y \to \alpha_{\rm tc}(u)).$$

Si suponemos $y, x_1, \ldots, x_n \in \omega$, en KP podemos probar las equivalencias siguientes:

$$\gamma_{\rm tc} \leftrightarrow \bigwedge u(u \in \omega \wedge u \le y \to \alpha^*(u)) \leftrightarrow \bigwedge u \in y\, \alpha^*_{\rm tc}(u) \wedge \alpha^*(y),$$

ya que $x \le y$ equivale a $x \in y \vee x = y$, luego basta tomar

$$\gamma^* \equiv \bigwedge u \in y\, \alpha^*(u) \wedge \alpha^*(y),$$

pues esta fórmula es claramente de tipo Δ_1 en KP.

Finalmente, en el caso $\gamma \equiv \bigvee u \le y\, \alpha(u)$ se razona análogamente que basta tomar

$$\gamma^* \equiv \bigvee u \in y\, \alpha^*(u) \vee \alpha^*(y).$$

Si ahora partimos de una fórmula de tipo Σ_1, digamos $\gamma \equiv \bigvee u\, \alpha$, con $\alpha(x)$ de tipo Δ_0, basta tomar $\gamma^* \equiv \bigvee u(u \in \omega \wedge \alpha^*(u))$, que es una fórmula Σ_1 en la jerarquía de Lévy y claramente cumple lo pedido. Igualmente se razona si γ es de tipo Π_1. ∎

Finalmente, si tenemos un caso de Σ_1-inducción en $\mathcal{L}_a$:

$$\gamma \equiv \phi(0) \wedge \bigwedge u\ (\phi(u) \to \phi(u')) \to \bigwedge u\, \phi(u),$$

donde la fórmula ϕ es Σ_1 y tiene variables libres $x, x_1, \ldots, x_n$, su traducción es

$$\gamma_{\rm tc} \equiv \phi_{\rm tc}(0) \wedge \bigwedge u \in \omega\ (\phi_{\rm tc}(u) \to \phi_{\rm tc}(u')) \to \bigwedge u \in \omega\, \phi_{\rm tc}(u).$$

Si suponemos $x_1, \ldots, x_n \in \omega$, tenemos que esto equivale a

$$\gamma^* \equiv \phi^*(0) \wedge \bigwedge u \in \omega\ (\phi^*(u) \to \phi^*(u')) \to \bigwedge u \in \omega\, \phi^*(u),$$

y, como ϕ^* es Σ_1, sabemos que γ^* es un teorema de KP. Por lo tanto:

[10] Técnicamente el argumento se formaliza en ARP como se indica en la nota en la prueba del teorema 3.25.

Teorema 5.45 KP *interpreta a* $\mathrm{I}\Sigma_1$.

Así pues, todos los teoremas aritméticos que hemos demostrado en ARP o en $\mathrm{I}\Sigma_1$ son válidos para los números naturales definidos en KP.

Ahora recordamos que en 5.40 hemos demostrado que $\mathrm{I}\Sigma_1$ interpreta a KP. En particular, podemos considerar la traducción a $\mathcal{L}_a$ de la fórmula $x \in \omega$ de $\mathcal{L}_{\mathrm{tc}}$, a la que seguiremos llamando $x \in \omega$. Los números naturales que cumplen $x \in \omega$ son los que, vistos como conjuntos, son ordinales.

Por ejemplo, como en KP se prueba que $0 = \varnothing \in \omega$, y en $\mathrm{I}\Sigma_1$ también se tiene que $0 = \varnothing$, podemos afirmar que $0 \in \omega$. Similarmente, en KP se prueba que $1 = \{0\} \in \omega$, y en $\mathrm{I}\Sigma_1$ también se prueba que $1 = \{0\}$, luego en $\mathrm{I}\Sigma_1$ tenemos que $1 \in \omega$, pero para 2 ya no es cierto. En efecto, en $\mathrm{I}\Sigma_1$ tenemos que $2 = \{1\}$ y $\{0, 1\} = 2^0 + 2^1 = 3$, por lo que el número natural 3 es el ordinal 2, mientras que $2 = \{1\} \notin \omega$ (no es un conjunto transitivo).

Más precisamente, consideremos la traducción del término $x' \equiv x \cup \{x\}$ de $\mathcal{L}_{\mathrm{tc}}$, al que seguiremos representando igual. Como x' es Δ_0 en KP, sabemos que es Δ_1 en $\mathrm{I}\Sigma_1$, y no es sino la traducción del término Δ_0 de $\mathcal{L}_{\mathrm{arp}}$ que llamamos igual.[11] Más aún, en ARP podemos definir un funtor mediante:

$$f(0) = 0, \qquad f(n+1) = f(n)' \equiv f(n) \cup \{f(n)\}$$

el cual se traduce a un término Δ_1 de $\mathcal{L}_a$ (o, alternativamente, el teorema 4.47 nos permite construir la fórmula $y = f(n)$). Si llamamos $\bar{n} \equiv f(n)$, tenemos, por ejemplo, que

$$\bar{0} = 0, \quad \bar{1} = \{\bar{0}\} = 3, \quad \bar{2} = \{\bar{0}, \bar{1}\} = \{0, 3\} = 9, \quad \ldots$$

y éstos son los números naturales x que cumplen $x \in \omega$. En efecto:

Teorema 5.46 *Si representamos por* $x +_a y$ *y* $x \cdot_a y$ *las traducciones de los términos* $x + y$ *y* $x \cdot y$ *de* $\mathcal{L}_{\mathrm{tc}}$ *(ambos son términos* Δ_1*), se cumple que*

1. $\bigwedge n\ \bar{n} \in \omega$,
2. $\bigwedge x(x \in \omega \to \bigvee n\ x = \bar{n})$,
3. $\bigwedge mn(\bar{m} = \bar{n} \to m = n)$,
4. $\bigwedge mn(\overline{m+n} = \bar{m} +_a \bar{n})$,
5. $\bigwedge mn(\overline{m \cdot n} = \bar{m} \cdot_a \bar{n})$.

Demostración: 1) Se cumple que $0 \in \omega \wedge \bigwedge x(x \in \omega \to x' \in \omega)$ es un teorema de KP, luego su traducción lo es de $\mathrm{I}\Sigma_1$. Por lo tanto, tenemos que $\bar{0} \in \omega$ así como que

$$\bar{n} \in \omega \to \overline{n+1} \in \omega,$$

luego por Σ_1-inducción concluimos $\bigwedge n, \bar{n} \in \omega$.

[11] Más precisamente, si llamamos x^* a la traducción del término $x \cup \{x\}$ de $\mathcal{L}_{\mathrm{arp}}$, no podemos afirmar que $x^* \equiv x'$, pero sí que $\underset{\mathrm{ARP}}{\vdash} x^* = x'$, pues tenemos $\bigwedge u(u \in x^* \leftrightarrow u \in x \vee u = x)$ y $\bigwedge u(u \in x' \leftrightarrow u \in x \vee u = x)$, luego son iguales por extensionalidad.

Para probar 2) razonamos por inducción completa sobre x, es decir, suponemos el resultado cierto para todo $y < x$. Entonces, si $x \in \omega$, o bien $x = 0$, en cuyo caso $x = \bar{0}$ y se cumple lo pedido, o bien $x \neq 0$, pero sabemos que

$$\bigwedge x \in \omega(x \neq 0 \to \bigvee y \in \omega(y \in x \wedge x = y')),$$

porque esto es la traducción de un teorema de KP. Así pues, existe un $y \in x$ tal que $y \in \omega \wedge x = y'$. La condición $y \in x$ implica $y < x$, luego por hipótesis de inducción existe un n tal que $y = \bar{n}$. Entonces $x = \bar{n}' = \overline{n+1}$ y se cumple lo pedido.

3) Probamos por inducción[12] sobre m que $\bigwedge n(\bar{m} = \bar{n} \to m = n)$. Si $m = 0$ y $\bar{m} = \bar{n}$, entonces $\bar{n} = \varnothing$ y esto implica que $n = 0$, pues si $n = k + 1$ entonces $\bar{n} = \bar{k} \cup \{\bar{k}\}$, luego $\bar{k} \in \bar{n} \neq \varnothing$.

Si vale para m y tenemos que $\overline{m+1} = \bar{n}$, entonces, según acabamos de ver, $\bar{m} \in \overline{m+1}$, luego $n \neq 0$, luego existe un k tal que $n = k+1$, luego $\overline{m+1} = \overline{k+1}$, es decir, $\bar{m}' = \bar{k}'$. Ahora usamos que $\bigwedge xy \in \omega\ (x' = y' \to x = y)$, porque es una traducción de un teorema de KP, luego $\bar{m} = \bar{k}$, luego $m = k$ por hipótesis de inducción, luego $m + 1 = n$.

4) Por inducción sobre n. Para $n = 0$ tenemos que $\overline{m+n} = \bar{m} = \bar{m} +_a \bar{0}$, donde hemos usado que $\bigwedge x \in \omega\ x +_a 0 = x$, por ser la traducción de un teorema de KP. Si vale para m, entonces

$$\overline{m+n+1} = \overline{m+n}' = (\bar{m} +_a \bar{n})' = \bar{m} +_a \bar{n}' = \bar{m} +_a \overline{n+1},$$

donde hemos usado que $\bigwedge xy \in \omega\ (x +_a y)' = x +_a y'$. La prueba de 5) es análoga. ∎

Ahora demostramos (en $\mathrm{I}\Sigma_1$) que los números naturales de $\mathrm{I}\Sigma_1$ tienen las mismas propiedades que los de KP:

Teorema 5.47 *Para toda fórmula $\phi(x_1, \ldots, x_n)$ de $\mathcal{L}_{\mathrm{a}}$ con las variables libres entre las indicadas, se cumple*

$$\underset{\mathrm{I}\Sigma_1}{\vdash} \phi(x_1, \ldots, x_n) \leftrightarrow (\phi_{\mathrm{tc}})_a(\bar{x}_1, \ldots, \bar{x}_n).$$

Para todo término $t(x_1, \ldots, x_n)$ de $\mathcal{L}_a$ con las variables libres entre las indicadas se cumple

$$\underset{\mathrm{I}\Sigma_1}{\vdash} \overline{t(x_1, \ldots, x_n)} = (t_{\mathrm{tc}})_a(\bar{x}_1, \ldots, \bar{x}_n).$$

Demostración: Probamos las dos afirmaciones por inducción[13] sobre la longitud de una expresión θ. Si $\theta \equiv x$ o $\theta \equiv 0$ es trivial. Si $\theta \equiv t'$, entonces por hipótesis de inducción

$$\overline{t(x_1, \ldots, x_n)} = (t_{\mathrm{tc}})_a(\bar{x}_1, \ldots, \bar{x}_n),$$

[12] La fórmula es Π_1.

[13] Como de costumbre, para formalizar la prueba en ARP construimos un funtor tal que $F(\theta, n)$ sea una función definida sobre el conjunto de las expresiones de longitud n que se obtienen de sustituir las variables ligadas que estén libres en una subsemiexpresión de θ por variables libres, y que asigne a cada una una demostración de la equivalencia del enunciado.

luego

$$\overline{t(x_1,\dots,x_n)}' = (t_{\mathrm{tc}})'_a(\bar{x}_1,\dots,\bar{x}_n),$$

pero $((t_{\mathrm{tc}})_a)' \equiv ((t_{\mathrm{tc}})')_a \equiv ((t')_{\mathrm{tc}})_a$, luego la fórmula anterior equivale a

$$\overline{t'(x_1,\dots,x_n)} = (t'_{\mathrm{tc}})_a(\bar{x}_1,\dots,\bar{x}_n).$$

Si $\theta \equiv t_1 + t_2$, por hipótesis de inducción

$$\overline{t_1(x_1,\dots,x_n)} = (t_{1\mathrm{tc}})_a(\bar{x}_1,\dots,\bar{x}_n) \wedge \overline{t_2(x_1,\dots,x_n)} = (t_{2\mathrm{tc}})_a(\bar{x}_1,\dots,\bar{x}_n),$$

luego, usando el teorema anterior,

$$\overline{t_1 + t_2} = \bar{t}_1 +_a \bar{t}_2 = (t_{1\mathrm{tc}})_a +_a (t_{2\mathrm{tc}})_a = ((t_1 + t_2)_{\mathrm{tc}})_a.$$

El caso $\theta \equiv t_1 \cdot t_2$ es análogo. Si $\theta \equiv t_1 = t_2$ tenemos que

$$(\theta_{\mathrm{tc}})_a(\bar{x}_1,\dots,\bar{x}_n) \equiv (t_{1\mathrm{tc}})_a = (t_{2\mathrm{tc}})_a \leftrightarrow \bar{t}_1 = \bar{t}_2 \leftrightarrow t_1 = t_2 \equiv \theta,$$

por el apartado 3) del teorema anterior.

Los casos $\theta \equiv \neg\phi$ o $\theta \equiv \phi \vee \psi$ son sencillos. Si $\theta \equiv \bigwedge u\ \phi(u, x_1,\dots,x_n)$, por hipótesis de inducción

$$\phi(x, x_1,\dots,x_n) \leftrightarrow (\phi_{\mathrm{tc}})_a(\bar{x}, \bar{x}_1,\dots,\bar{x}_n),$$

lo cual implica

$$\textstyle\bigwedge u\phi(u, x_1,\dots,x_n) \leftrightarrow \bigwedge u(\phi_{\mathrm{tc}})_a(\bar{u}, \bar{x}_1,\dots,\bar{x}_n).$$

Usando el apartado 2) del teorema anterior es fácil ver que la parte derecha equivale a

$$\textstyle\bigwedge u \in \omega\ (\phi_{\mathrm{tc}})_a(u, \bar{x}_1,\dots,\bar{x}_n),$$

que a su vez es lo mismo que $(\bigwedge u \in \omega\, \phi_{\mathrm{tc}})_a \equiv ((\bigwedge u\, \phi)_{\mathrm{tc}})_a$, luego tenemos que

$$\textstyle\bigwedge u\phi(x, x_1,\dots,x_n) \leftrightarrow ((\bigwedge u\, \phi)_{\mathrm{tc}})_a(\bar{u}, \bar{x}_1,\dots,\bar{x}_n).$$

El caso del particularizador es análogo.

Por último, si $\theta \equiv u|\phi(u, x_1,\dots,x_n)$, entonces $\theta_{\mathrm{tc}} \equiv u|(u \in \omega \wedge \phi_{\mathrm{tc}})$ y $(\theta_{\mathrm{tc}})_a \equiv u|(u \in \omega \wedge (\phi_{\mathrm{tc}})_a)$. Tenemos la misma hipótesis de inducción, de la que se sigue claramente que

$$\textstyle\bigvee^1 u\, \phi(u, x_1,\dots,x_n) \leftrightarrow \bigvee^1 u(\phi_{\mathrm{tc}})_a(\bar{u}, \bar{x}_1,\dots,\bar{x}_n).$$

Usando de nuevo los apartados 2) y 3) del teorema anterior es fácil ver que esto equivale a

$$\textstyle\bigvee^1 u\, \phi(u, x_1,\dots,x_n) \leftrightarrow \bigvee^1 u \in \omega(\phi_{\mathrm{tc}})_a(u, \bar{x}_1,\dots,\bar{x}_n).$$

Dados $x_1,\dots,x_n$, si se dan las dos partes de la equivalencia y $x = u|\phi(u)$, entonces $\phi(x, x_1,\dots,x_n)$, luego por hipótesis de inducción tenemos también

que $\bar{x} \in \omega \wedge (\phi_{\mathrm{tc}})_a(\bar{x}, \bar{x}_1, \ldots, \bar{x}_n)$, luego $\bar{x} = u|(u \in \omega \wedge (\phi_{\mathrm{tc}})_a(u))$, es decir, se cumple que $\bar{\theta} = (\theta_{\mathrm{tc}})_a$, como había que probar.

Si no se dan las unicidades, entonces $\bar{\theta} = (\theta_{\mathrm{tc}})_a$, pues ambas descripciones son impropias. ∎

En particular, si γ es una sentencia de $\mathcal{L}_a$, tenemos que

$$\underset{\mathrm{I}\Sigma_1}{\vdash} (\gamma \leftrightarrow (\gamma_{\mathrm{tc}})_a).$$

Así llegamos al teorema siguiente:

Teorema 5.48 *Si γ es una sentencia de $\mathcal{L}_a$, entonces*

$$\underset{\mathrm{I}\Sigma_1}{\vdash} \gamma \quad \textit{si y sólo si} \quad \underset{\mathrm{KP}}{\vdash} \gamma_{\mathrm{tc}}.$$

Demostración: Si $\underset{\mathrm{I}\Sigma_1}{\vdash} \gamma$, entonces $\underset{\mathrm{KP}}{\vdash} \gamma_{\mathrm{tc}}$ porque KP interpreta a $\mathrm{I}\Sigma_1$. Recíprocamente, si $\underset{\mathrm{KP}}{\vdash} \gamma_{\mathrm{tc}}$, entonces $\underset{\mathrm{I}\Sigma_1}{\vdash} (\gamma_{\mathrm{tc}})_a$ porque $\mathrm{I}\Sigma_1$ interpreta a KP y $\underset{\mathrm{I}\Sigma_1}{\vdash} \gamma$ por la observación tras el teorema anterior. ∎

Así pues, $\mathrm{I}\Sigma_1$ y KP permiten demostrar exactamente los mismos teoremas sobre números naturales. Esto significa que, en lugar de trabajar en $\mathrm{I}\Sigma_1$ pensando que los números naturales también son conjuntos, podemos trabajar en KP, donde los números naturales son sólo parte de los conjuntos, con la garantía de que cualquier teorema sobre números naturales que demostremos en KP será, de hecho, demostrable en $\mathrm{I}\Sigma_1$ y si, más precisamente, es expresable en $\mathcal{L}_{\mathrm{arp}}$ (lo que equivale a que se trate de una sentencia Π_1), entonces será demostrable en ARP.

Igualmente:

Teorema 5.49 *Si γ es una sentencia de $\mathcal{L}_a$, entonces*

$$\underset{\mathrm{AP}}{\vdash} \gamma \quad \textit{si y sólo si} \quad \underset{\mathrm{Z}^*}{\vdash} \gamma_{\mathrm{tc}}.$$

Demostración: Si $\underset{\mathrm{AP}}{\vdash} \gamma$, tenemos que $\underset{\mathrm{Z}^*}{\vdash} \gamma_{\mathrm{tc}}$ porque hemos visto que Z^* interpreta a AP (teorema 5.28). Recíprocamente, si $\underset{\mathrm{Z}^*}{\vdash} \gamma_{\mathrm{tc}}$, entonces $\underset{\mathrm{AP}}{\vdash} (\gamma_{\mathrm{tc}})_a$, porque AP interpreta Z^* (teorema 5.30). Por lo tanto $\underset{\mathrm{AP}}{\vdash} \gamma$ (pues la equivalencia $\gamma \leftrightarrow (\gamma_{\mathrm{tc}})_a$ es probable en $\mathrm{I}\Sigma_1$, luego en AP). ∎

Con esto queda demostrado que la aritmética de Z^* es exactamente la aritmética de Peano. Sin embargo, vamos a ver que podemos decir algo más fuerte:

La teoría de Zermelo-Fraenkel La teoría de conjuntos que consideró Zermelo constaba en realidad de dos axiomas más,[14] aparte de los que hemos incluido en Z^*. Uno es el axioma de infinitud (AI), que postula la existencia de

[14] En realidad eran tres, pues Zermelo incluyó entre sus axiomas el llamado axioma de elección, pero actualmente no se considera parte de la teoría de Zermelo.

un conjunto infinito, que en este contexto no nos interesa porque su traducción es falsa en $\mathrm{I}\Sigma_1$ o en AP. El segundo es el *axioma de partes* (AP):

$$\bigwedge x \bigvee y \bigwedge u(u \in y \leftrightarrow u \subset x).$$

Si a Z^* le añadimos el axioma de partes, tenemos la teoría que podemos llamar $\mathrm{Z} - \mathrm{AI}$ (la teoría de Zermelo sin el axioma de infinitud). Por el axioma de extensionalidad, el conjunto y cuya existencia afirma AP es único, luego podemos definir

$$\mathcal{P}x \equiv y \mid \bigwedge u(u \in y \leftrightarrow u \subset x).$$

de modo que en $\mathrm{Z} - \mathrm{AI}$ tenemos $\bigwedge u(u \in \mathcal{P}x \leftrightarrow u \subset x)$.

El interés principal de este axioma en nuestro contexto es que nos permite definir productos cartesianos. En efecto, de acuerdo con la definición que hemos dado de par ordenado: $(x, y) \equiv \{\{x\}, \{x, y\}\}$, si tenemos que $x, y \in z$, se cumple que $\{x\}, \{x, y\} \in \mathcal{P}z$, luego $(x, y) \in \mathcal{P}\mathcal{P}z$. Por lo tanto, podemoss definir

$$x \times y \equiv \{z \in \mathcal{P}\mathcal{P}(x \cup y) \mid \bigvee u \in x \bigvee v \in y\, z = (u, v)\},$$

y en $\mathrm{Z} - \mathrm{AI}$ se demuestra que

$$z \in x \times y \leftrightarrow \bigvee u \in x \bigvee v \in y\, z = (u, v).$$

Aunque la mayor parte de las matemáticas modernas se pueden formalizar en la teoría ZC (Z más el axioma de elección, que es en realidad la teoría original de Zermelo), una parte requiere dos axiomas más, que al añadirlos a Z dan lugar a la teoría de *Zermelo-Fraenkel* (ZF). Estos axiomas son el *axioma de reemplazo* (que es en realidad un esquema axiomático y que de hecho implica el axioma de especificación, por lo que éste se vuelve redundante) y el *axioma de regularidad* (véase la tabla siguiente):

Axiomas de ZF − AI

Extensionalidad	$\bigwedge xy(\bigwedge u(u \in x \leftrightarrow u \in y) \to x = y)$
Par	$\bigwedge xy \bigvee z(x \in z \wedge y \in z)$
Unión	$\bigwedge x \bigvee y \bigwedge u \in x \bigwedge v \in u\ v \in y$
Partes	$\bigvee y \bigwedge u(u \in y \leftrightarrow u \subset x)$
Reemplazo	$\bigwedge xyz(\phi(x, y) \wedge \phi(x, z) \to y = z) \to$ $\bigwedge a \bigvee b \bigwedge y(y \in b \leftrightarrow \bigvee x \in a\, \phi(x, y))$.
Regularidad	$\bigvee v\, v \in y \to \bigvee v(v \in y \wedge \neg \bigvee u(u \in v \wedge u \in y))$

Finalmente tenemos el axioma de elección, que admite muchas formulaciones equivalentes, por ejemplo que todo conjunto admite un buen orden, es decir, una relación de orden en la que todo subconjunto no vacío tiene un mínimo elemento. Sabemos que esto se cumple en ARP, luego en $\mathrm{I}\Sigma_1$, luego en AP, ya que en ARP todo conjunto no vacío tiene un mínimo elemento respecto a la relación de orden usual de los números naturales.

Cuando añadimos el axioma de elección a Z o a ZF obtenemos las teorías ZC y ZCF, respectivamente. Vamos a ver que, sin el axioma de infinitud, son interpretables en AP:

Teorema 5.50 AP *interpreta a* ZFC$-$AI.

DEMOSTRACIÓN: Acabamos de señalar que el axioma de elección es un teorema de $\mathrm{I}\Sigma_1$, y ya hemos visto que AP interpreta a Z^*, luego sólo nos falta comprobar las traducciones de los axiomas de partes, reemplazo y regularidad.

Tras el teorema 2.16 hemos visto que el conjunto de partes $\mathcal{P}x$ puede definirse en ARP, luego en $\mathrm{I}\Sigma_1$ se prueba su existencia. Por lo tanto, la traducción del axioma de partes a $\mathcal{L}_a$ es un teorema de $\mathrm{I}\Sigma_1$.

Para probar la traducción del axioma de reemplazo suponemos

$$\bigwedge xyz(\phi(x,y) \wedge \phi(x,z) \to y = z)$$

y consideramos la fórmula $\psi(x,y) \equiv \phi(x,y) \vee (\neg\bigvee u\, \phi(x,u) \wedge y = 0)$. Es claro que $\bigwedge x < a \overset{1}{\bigvee} y\, \psi(x,y)$, luego por el principio de recolección 4.25 existe un v tal que $\bigwedge x < a \bigvee y < v\, \psi(x,y)$. A continuación tomamos

$$b = \{y \in I_v \mid \bigvee x \in a\, \phi(x,y)\},$$

(donde I_v es el conjunto de los números naturales menores que v). De este modo, si $\bigvee x \in a\, \phi(x,y)$, existe un $y' < v$ tal que $\psi(x,y')$, pero por definición de ψ tiene que ser $\phi(x,y')$, y por la unicidad $y = y' \in I_v$, luego $y \in b$.

La traducción del axioma de regularidad a $\mathcal{L}_a$ es trivialmente demostrable en ARP, pues basta tomar como v el mínimo de los elementos de y. Esto hace que también sea demostrable en $\mathrm{I}\Sigma_1$. ∎

Esto tiene una consecuencia notable, y es que si en la prueba de 5.49 usamos el teorema precedente en lugar de 5.30 obtenemos lo siguiente:

Teorema 5.51 *Si γ es una sentencia de $\mathcal{L}_a$, entonces*

$$\underset{\mathrm{AP}}{\vdash} \gamma \quad \textit{si y sólo si} \quad \underset{\mathrm{ZFC-AI}}{\vdash} \gamma_{\mathrm{tc}}.$$

Así pues, en ZFC$-$ AI podemos demostrar exactamente los mismos teoremas sobre números naturales que en Z^*, ni más ni menos que los teoremas de la aritmética de Peano. Los axiomas adicionales permiten demostrar más cosas sobre conjuntos, pero no más teoremas sobre números naturales.

Nota Por el mismo argumento, dado que en $\mathrm{I}\Sigma_1$ se demuestra la traducción del axioma de partes, en el teorema 5.48 podemos sustituir KP por KP + AP. ∎

5.6 Conjuntos finitos, cardinales

En esta sección ilustraremos cómo se trabaja en KP demostrando las propiedades básicas de los conjuntos finitos. Conviene observar que, aunque la teoría Z − AI no es una extensión de KP, todos los argumentos valen también trivialmente en esta teoría. La razón es que, aunque por caminos muy distintos, en ambas teorías hemos demostrado la existencia del producto cartesiano de dos conjuntos dados, y a partir de ahí podemos definir por especificación en Z − AI todos los conceptos relacionados con aplicaciones (dominio, rango, composición, etc.) que vamos a necesitar. La definición de finitud es la obvia:

Definición 5.52 x es finito $\equiv \bigvee f \bigvee n \in \omega\ f : n \longrightarrow x$ biyectiva.

Notemos que en KP cada número natural n coincide con el conjunto I_n de los números naturales anteriores a él, por lo que no podemos poner n donde en ARP poníamos I_n.

El teorema siguiente implica que el número natural n está unívocamente determinado:

Teorema 5.53 $\bigwedge mn \in \omega \bigwedge f(f : m \longrightarrow n$ *inyectiva* $\to m \leq n)$.

Demostración: Razonamos por inducción[15] sobre m. Si $m = 0$ es trivial. Supuesto cierto para m, supongamos que $f : (m+1) \longrightarrow n$ inyectiva. Podemos suponer que $f(m) = n - 1$. En efecto, si $n - 1 \notin \mathcal{R}f$, basta tomar

$$f^* = (f \setminus \{(m, f(m))\}) \cup \{(m, n-1)\},$$

y es claro que $f^* : (m+1) \longrightarrow n$ inyectiva y $f^*(m) = n - 1$. Si $n - 1 \in \mathcal{R}f$, pero no es $f(m)$, sea $u < m$ tal que $f(u) = n - 1$. Tomamos

$$f^* = (f \setminus \{(m, f(m)), (u, n-1)\}) \cup \{(u, f(m), (m, n-1)\},$$

y es claro que esta f^* cumple lo mismo que en el caso anterior.

Pero entonces, $f \setminus \{(m, n-1)\} : m \longrightarrow (n-1)$ inyectiva, luego por hipótesis de inducción $m \leq n - 1$, luego $m + 1 \leq n$. ∎

En particular,

$$\bigwedge nmf(f : m \longrightarrow n \text{ biyectiva} \to m = n),$$

luego podemos definir el *cardinal* de un conjunto finito:

Definición 5.54 $|x| \equiv n | \bigvee f\ f : n \longrightarrow x$ *biyectiva.*

[15] La fórmula es claramente Π_1. Recordemos que disponemos tanto de la inducción Σ_1 como de la inducción Π_1.

El resultado básico sobre cardinales es el siguiente:

Teorema 5.55 *Si x e y son conjuntos finitos, se cumple:*

1. $\bigwedge xy(|x| = |y| \leftrightarrow \bigvee f\ f : x \longrightarrow y$ *biyectiva*),
2. $\bigwedge xy(|x| \leq |y| \leftrightarrow \bigvee f\ f : x \longrightarrow y$ *inyectiva*),

Demostración: 1) Si $|x| = |y| = n$, existen $g : n \longrightarrow x$, $h : n \longrightarrow y$ biyectivas, y basta tomar $f = g^{-1} \circ h$. Si $f : x \longrightarrow y$ biyectiva y $|x| = n$, entonces existe $g : n \longrightarrow x$ biyectiva, luego $g \circ f : n \longrightarrow y$ biyectiva, luego $|y| = n = |x|$.

2) Sean $|x| = m$, $|y| = n$, $g : m \longrightarrow x$ biyectiva y $h : n \longrightarrow y$ biyectiva. Si $m \leq n$ entonces $m \subset n$, luego $g^{-1} \circ h : x \longrightarrow y$ inyectiva. Recíprocamente, si $f : x \longrightarrow y$ inyectiva, $g \circ f \circ h^{-1} : m \longrightarrow n$ es inyectiva, luego $m \leq n$ por el teorema anterior. ∎

Ahora caracterizamos la suma y el producto en términos de cardinales. La suma es el cardinal de la unión disjunta:

Teorema 5.56 *Si x, y son conjuntos finitos y $x \cap y = \varnothing$, entonces $x \cup y$ es finito y $|x \cup y| = |x| + |y|$).*

Demostración: Fijemos $f : x \longrightarrow |x|$, $g : y \longrightarrow |y|$ biyectivas, y consideremos el conjunto definido por Δ_1-especificación:

$$h = \{(u, v) \in x \times (|x|+|y|) \mid v = f(u)\} \cup \{(u, v) \in y \times (|x|+|y|) \mid v = |x|+g(u)\}.$$

Es fácil ver que[16] $h : x \cup y \longrightarrow |x| + |y|$ biyectiva. ∎

Teorema 5.57 *Si x e y son conjuntos finitos, entonces $|x \times y| = |x| \cdot |y|$.*

Demostración: La prueba es análoga a la anterior, tomando ahora

$$h = \{(u, v) \in (x \times y) \times |x| \cdot |y| \mid \bigvee rs \in \bigcup\bigcup(x \cup y)(u = (r, s) \wedge v = |y| \cdot f(r) + g(s))\}.$$

∎

Teorema 5.58 *Si x es un conjunto finito y $u \subset x$, entonces u es finito y cumple $|u| \leq |x|$. Además, se cumple $|u| = |x|$ si y sólo si $u = x$.*

Demostración: Sea $|x| = n$. Entonces u se puede biyectar con un $v \subset n$, y basta ver que v es finito con $|v| \leq n$. Veamos por inducción sobre m que[17]

$$\bigvee f \bigvee r \leq m\ f : r \longrightarrow v \cap m \text{ biyectiva.}$$

[16] Recordemos que en KP podemos usar todos los teoremas aritméticos demostrables en $I\Sigma_1$, como que si $v \in \omega$ cumple $|x| \leq v < |x| + |y|$ existe un $u < |y|$ tal que $v = |x| + u$.

[17] La fórmula es Σ_1, por lo que la inducción es lícita.

Si $m = 0$ es trivial. Si vale para m, o bien $m \notin v$, en cuyo caso tenemos que $v \cap m = v \cap (m+1)$ y la conclusión es obvia, o bien $m \in v$, en cuyo caso tenemos que $v \cap (m+1) = (v \cap m) \cup \{m\}$ y la unión es disjunta. Por hipótesis de inducción $v \cap m$ es finito y $|v \cap m| \leq m$, luego

$$|v \cap (m+1)| = |v \cap m| + 1 \leq m + 1.$$

En particular, aplicando esto a $m = n$ tenemos que $|v| \leq m$.

Si $|u| = |x|$ entonces $x = u \cup (x \setminus u)$, luego $|x| = |u| + |x \setminus u|$, luego $|x \setminus u| = 0$, luego $x \setminus u = \varnothing$, luego $u = x$. ∎

Teorema 5.59 *Si x e y son conjuntos finitos, entonces $x \cup y$ es finito y además $|x \cup y| \leq |x| + |y|$.*

Demostración: $x \cup y = x \cup (y \setminus x)$ es una unión de conjuntos finitos disjuntos, luego es finita. Además $|x \cup y| = |x| + |y \setminus x| \leq |x| + |y|$. ∎

Más en general:

Teorema 5.60 *Si a es un conjunto finito y $\bigwedge x \in a\ x$ es finito, entonces $\bigcup a$ es finito.*

Demostración: Sea $f : n \longrightarrow a$ biyectiva. Basta probar por inducción sobre m que $m \leq n \to \bigcup f[m]$ es finito.

Como ilustración, vamos a detallar la justificación de que la fórmula es Σ_1, lo cual es necesario para que la inducción sea lícita.

Partimos de que "x es finito" es Σ_1, pues equivale a

$$\bigvee n f(n \in \omega \wedge f : x \longrightarrow n \text{ biyectiva}),$$

y la fórmula tras los cuantificadores es Δ_0. Ahora usamos que el término $\bigcup x$ es Δ_0 (está probado en la tabla de la página 241), luego al sustituir x por $\bigcup x$ obtenemos una fórmula[18] Σ_1, que en este caso es "$\bigcup x$ es finito". El término $f[m]$ también es Δ_0 (véase la tabla de la página 258), luego al sustituir x por $f[m]$ obtenemos que "$\bigcup f[m]$ es finito" es Σ_1. Como $m \leq n$ es Δ_0 (es la inclusión) la implicación es Σ_1.

Si $m = 0$, entonces $\bigcup f[0] = \varnothing$ es finito. Si vale para m y $m + 1 \leq n$, entonces $f[m+1] = f[m] \cup \{f(m)\}$, luego $\bigcup f[m+1] = \bigcup f[m] \cup f(m)$. El primer conjunto es finito por hipótesis de inducción y el segundo por hipótesis, luego la unión es finita. ∎

[18] Recordamos el argumento:

$$\bigcup x \text{ es finito} \leftrightarrow \bigvee u(u = \bigcup x \wedge u \text{ es finito}).$$

Teorema 5.61 *y es finito* $\to \overset{1}{\bigvee} z \bigwedge s(s \in z \leftrightarrow s : y \longrightarrow x)$.

DEMOSTRACIÓN: Lo probaremos en primer lugar en el caso en que y es un número natural. Vamos a usar el teorema[19] 5.41 para la fórmula Δ_1

$$\psi(x,n,a,z) \equiv \bigwedge f \in z\,(f : n+1 \longrightarrow x \wedge f|_n \in a) \wedge$$

$$\bigwedge s \in a \bigwedge u \in x (s : n \longrightarrow x \to \bigvee f \in z\,(f|_n = s \wedge f(n) = u)).$$

Vamos a probar que $\bigwedge n \in \omega \overset{1}{\bigvee} z\,\psi(x,n,a,z)$. Dado un conjunto a, consideramos el subconjunto

$$b = \{s \in a \mid s : n \longrightarrow x\},$$

que existe por Δ_1-especificación. Consideramos también la fórmula Δ_1

$$\phi(f,s,u) \equiv f : n+1 \longrightarrow x \wedge f|_n = s \wedge f(n) = u$$

y observamos que $\bigwedge s \in b \bigwedge u \in x \overset{1}{\bigvee} f\,\phi(f,s,u)$, luego por Σ_1-reemplazo 5.39 existe una aplicación $h : b \times x \longrightarrow z$ suprayectiva tal que

$$\bigwedge s \in b \bigwedge u \in x\ \phi(s,x,h(s,x)).$$

Claramente z es el conjunto buscado. La unicidad se sigue del axioma de extensionalidad, pues z contiene exactamente a las sucesiones en x que extienden a sucesiones de a.

Aplicando el teorema 5.41 a $G(x,a,n) \equiv z|\psi(x,n,a,z)$ obtenemos una fórmula $\chi(x,n,z)$ de tipo Δ_1 de modo que $\bigwedge n \in \omega \overset{1}{\bigvee} z\,\chi(x,n,z)$ y, si llamamos $x^n \equiv z|\chi(x,n,z)$, se cumple

$$x^0 = \{\varnothing\} \wedge \bigwedge n \in \omega\ x^{n+1} = G(x,n,x^n).$$

Ahora bien, si $f : n \longrightarrow x$ es cualquier aplicación, podemos probar por Σ_1-inducción que

$$\bigwedge i \le n \bigvee z(z = x^i \wedge \bigvee g \in z\ g = f|_i)).$$

En particular, $f \in x^n$, de modo que $z = x^n$ cumple lo requerido por el enunciado cuando $y = n$.

Si y es un conjunto finito arbitrario, existe $n \in \omega$ y una aplicación $f : y \longrightarrow n$ biyectiva. Basta aplicar Σ_1-reemplazo a la fórmula $\phi(t,s) \equiv s = f \circ t$ y al conjunto $a = x^n$, con lo que obtenemos un conjunto z tal que

$$\bigwedge s(s \in z \leftrightarrow \bigvee t \in x^n\ s = f \circ t).$$

Es fácil ver que z cumple lo pedido, y es único por extensionalidad. ∎

[19] Una prueba por inducción sobre el cardinal de y no sería técnicamente viable porque la fórmula correspondiente no es Σ_1 ni Π_1. En Z − AI sí que podemos razonar fácilmente por inducción.

Definición 5.62 $x^y \equiv z \mid \bigwedge s(s \in z \leftrightarrow s : y \longrightarrow x)$.

Acabamos de probar que x^y es una descripción propia siempre que y es finito.

Teorema 5.63 *Si x e y son finitos, entonces x^y es finito y $|x^y| = |x|^{|y|}$.*

DEMOSTRACIÓN: Sean $|x| = m$, $|y| = n$. Entonces es fácil definir una biyección entre[20] x^y y ${}^n m$ (de hecho, la hemos definido en la prueba del teorema anterior). Por lo tanto, basta probar que ${}^n m$ es finito y que $|{}^n m| = m^n$. Probamos por inducción[21] sobre n que $\bigvee f\ f : m^n \longrightarrow {}^n m$ biyectiva.

Para $n = 0$ es claro. Si se cumple para n, usamos que podemos construir una biyección de ${}^{n+1}m$ en ${}^n m \times m$ mediante $s \mapsto (s|_n, s(n))$, con lo que si ${}^n m$ es finito, también lo es ${}^{n+1}m$ y además

$$|{}^{n+1}m| = |{}^n m| \cdot m = m^n \cdot m = m^{n+1}.$$

■

Teorema 5.64 *x es finito $\rightarrow \overset{1}{\bigvee} y \bigwedge u(u \in y \leftrightarrow u \subset x)$.*

DEMOSTRACIÓN: Basta aplicar Σ_1-reemplazo al conjunto 2^x y a la fórmula $\phi(s, u) \equiv u = s^{-1}[1]$. Así obtenemos un conjunto y tal que

$$\bigwedge u(u \in y \leftrightarrow \bigvee s(s : x \longrightarrow \{0, 1\} \wedge u = s^{-1}[1])).$$

es fácil ver que y cumple lo pedido, y la unicidad se tiene por el axioma de extensionalidad. ■

Definición 5.65 $\mathcal{P}x \equiv y | \bigwedge u(u \in y \leftrightarrow u \subset x)$.

Acabamos de probar que $\mathcal{P}x$ es una descripción propia siempre que x es finito.

Precisando ligeramente la prueba del teorema anterior vemos que la aplicación $2^x \longrightarrow \mathcal{P}x$ allí construida es biyectiva. Por lo tanto:

Teorema 5.66 *Si x es un conjunto finito, entonces $\mathcal{P}x$ también es finito, y $|\mathcal{P}x| = 2^{|x|}$.*

Terminamos probando algunas propiedades básicas adicionales de los conjuntos finitos.

El concepto de relación de orden en un conjunto se define exactamente igual que en $\mathrm{I}\Sigma_1$ (notemos que la definición es claramente Δ_0):

Definición 5.67 Un conjunto R es una *relación de orden* en un conjunto x si cumple:

[20] Es costumbre escribir ${}^n m$ en lugar de m^n para referirse al conjunto de todas las aplicaciones de n en m cuando es posible confundir m^n con otra operación, como en este caso la exponenciación de números naturales.

[21] Notemos que la fórmula es Σ_1.

1. $R \subset x \times x$ (y escribiremos $u\,R\,v \equiv (u,v) \in R$).
2. $\bigwedge u \in x\, u\,R\,u$,
3. $\bigwedge uv \in x(u\,R\,v \wedge v\,R\,u \to u = v)$,
4. $\bigwedge uvw \in x(u\,R\,v \wedge v\,R\,w \to u\,R\,w)$.

   La relación es de *orden total* si además cumple:

5. $\bigwedge uv \in x(u\,R\,v \vee v\,R\,u)$.

Si R es una relación de orden en x, se dice que

1. u es un *R-minimal* de x si $u \in x \wedge \bigwedge v \in x\, \neg v\,R\,u$.
2. u es un *R-maximal* de x si $u \in x \wedge \bigwedge v \in x\, \neg u\,R\,v$.
3. u es un *R-mínimo* de x si $u \in x \wedge \bigwedge v \in x\, u\,R\,v$.
4. u es un *R-máximo* de x si $u \in x \wedge \bigwedge v \in x\, v\,R\,u$.

Es claro que si un conjunto ordenado tiene máximo o mínimo, éste es único.

Teorema 5.68 *Sea $x \neq \varnothing$ un conjunto finito. Entonces:*

1. *x admite un buen orden, es decir un orden total en el que todo subconjunto no vacío tiene mínimo.*
2. *Toda relación de orden en x tiene un elemento maximal y un minimal.*
3. *Toda relación de orden total en x es un buen orden.*

Demostración: 1) Sea $f : x \longrightarrow n$ biyectiva. Definimos (por Δ_1-especificación)

$$R = \{(u,v) \in x \times x \mid f(u) \leq f(v)\}.$$

Claramente es un buen orden en x.

2) Sea $f : n \longrightarrow x$ biyectiva. Si R es una relación de orden en x, definimos (por Δ_1-especificación)

$$R' = \{(u,v) \in n \times n \mid f(u)\,R\,f(v)\}$$

y así R' es una relación de orden en n, y basta ver que n tiene maximal respecto a R' (para concluir que tiene minimal aplicamos el resultado a R'^{-1}). Para ello probamos por inducción sobre m que si $1 \leq m \leq n$ entonces m tiene R'-maximal.[22] Obviamente 0 es R' maximal para 1. Si m tiene R'-maximal, digamos u, entonces, o bien $\neg u\,R'\,m$, en cuyo caso u es R'-maximal de $m+1$, o bien $u\,R'\,m$, en cuyo caso m es un R'-maximal de $m+1$. Así pues, n tiene R-maximal.

3) Si $y \subset x$ es un subconjunto no vacío, entonces es finito, luego tiene R-minimal, pero un R-minimal para una relación de orden total es un mínimo. ∎

[22] Explícitamente $\bigvee u < m \bigwedge v < m\,(v \neq u \to (u,v) \notin R')$, que es Δ_0.

Teorema 5.69 *Si $f : x \longrightarrow x$ y x es finito, entonces f es inyectiva si y sólo si es suprayectiva.*

DEMOSTRACIÓN: Podemos suponer que $x \neq \varnothing$. Si f es inyectiva, entonces $f[x] \subset x$ cumple $|f[x]| = |x|$, luego $f[x] = x$ y f es suprayectiva.

Si f es suprayectiva, fijamos un buen orden R en x. Entonces podemos definir $g : x \longrightarrow x$ inyectiva que a cada $u \in x$ le asigne su menor antiimagen. Concretamente:

$$g = \{(u, v) \in x \times x \mid f(v) = u \wedge \bigwedge w \in x(f(w) = u \to v\, R\, w)\},$$

que existe por Δ_1-especificación. Por la parte ya probada, g es biyectiva, luego f es inyectiva, ya que si $f(v) = f(w)$, entonces $g(v) = g(w)$, luego $v = w$. ∎

5.7 $I\Sigma_1$ como teoría de conjuntos

En la sección 5.5 hemos probado que KP es equivalente a $I\Sigma_1$ como teoría aritmética, en el sentido de que en ambas teorías se pueden demostrar los mismos resultados aritméticos. Ahora vamos a probar que $I\Sigma_1$ como teoría de conjuntos es equivalente a la teoría $\mathrm{KP}_{\mathrm{fin}}$, que resulta de añadir a KP el axioma "todo conjunto es finito".

Observemos que la traducción a $I\Sigma_1$ de "todo conjunto es finito" es

$$\bigwedge x \bigvee f \bigvee n \in \omega, f : n \longrightarrow x \text{ biyectiva.}$$

Vamos a ver que esto es un teorema de $I\Sigma_1$. Con la notación previa al teorema 5.46, esto equivale a

$$\bigwedge x \bigvee f n\ f : \bar{n} \longrightarrow x \text{ biyectiva.}$$

Por otra parte, en virtud del teorema 2.20, en $I\Sigma_1$ se puede probar que todo conjunto es finito en el sentido de que

$$\bigvee k \bigvee f\ f : I_k \longrightarrow x \text{ biyectiva.}$$

Pero, aplicando este mismo teorema a $\bar{n}$, vemos que existe[23] un k y una aplicación biyectiva $I_k \longrightarrow \bar{n}$, de donde también hay una aplicación $g : \bar{n} \longrightarrow x$ biyectiva. Así pues:

Teorema 5.70 $I\Sigma_1$ *interpreta a* $\mathrm{KP}_{\mathrm{fin}}$.

Por lo tanto, si γ es un teorema de $\mathrm{KP}_{\mathrm{fin}}$, su traducción γ_a es un teorema conjuntista de $I\Sigma_1$. Nos proponemos demostrar el recíproco. De momento trabajamos en KP. Necesitamos una variante del teorema 5.41:

[23] Aunque no lo necesitamos aquí, una simple inducción demuestra que $|\bar{n}| = n$, con lo que el realidad $k = n$.

Teorema 5.71 *Sea $G : \omega \times V \longrightarrow V$ una función Σ_1, Entonces existe una función $F : \omega \longrightarrow V$ de tipo Σ_1 tal que*

$$\bigwedge n \in \omega\ F(n) = G(n, F|_n),$$

donde $F|_n = \{(i, x) \mid i < n \wedge x = F(i)\}$.

La prueba es análoga a la de 5.26, tomando ahora

$$F = \{(n, x) \in \omega \times V \mid \bigvee s(s : n' \longrightarrow V \wedge s(n) = x \wedge \bigwedge i < n'\ s(i) = G(i, s|_i))\}.$$

Es fácil ver que si $F(n) = x$ y s cumple la definición de F, entonces $s = F|_n$, de donde se sigue la afirmación del enunciado (y en particular esto prueba que $F|_n$ es un conjunto).

Definimos la *pertenencia aritmética* como la traducción $x \in_a y$ a $\mathcal{L}_{\mathrm{tc}}$ (en el sentido de 5.19) de la fórmula $x \in y$ de $\mathcal{L}_a$ definida en $\mathrm{I}\Sigma_1$. Por 5.44, sabemos que $x \in_a y$ es una fórmula Δ_1 en KP. Aplicamos el teorema anterior a la clase G definida por la fórmula Σ_1

$$\psi(n, s, y) \equiv \bigwedge u \in y \bigvee i \in n(i \in_a n \wedge s(i) = u) \wedge \bigwedge i \in n(i \in_a n \to s(i) \in y).$$

Notemos que $\overset{1}{\bigvee} y\, \psi(n, s, i)$, luego ψ define ciertamente una función G y obtenemos otra función F de tipo Σ_1 ta que $c_n \equiv F(n)$ es un término de tipo Δ_1 que cumple

$$\bigwedge n \in \omega\ c_n = \{c_i \mid i \in_a n\}.$$

Teorema 5.72 *Se cumple:*

1. $\bigwedge ij \in \omega(c_i = c_j \leftrightarrow i = j)$.
2. $\bigwedge ij \in \omega(c_i \in c_j \leftrightarrow i \in_a j)$.

Demostración: De la propia definición de c_i se sigue que

$$\bigwedge in \in \omega(i \in_a n \to c_i \in c_n).$$

Para probar 1) basta ver por inducción[24] sobre n que

$$\bigwedge ij \in n(c_i = c_j \to i = j).$$

En efecto, para $n = 0$ es trivial, y si vale para n, supongamos que existe un $i < n$ tal que $c_i = c_n$. El hecho de que $i < n$ implica que $n \not\subset_a i$, es decir, que existe un $j \in_a n$ tal que $j \notin_a i$. Entonces $c_j \in c_n = c_i$, luego $c_j = c_k$ para cierto $k \in_a i$, pero esto implica $k < i < n$, luego $j = k$, luego $j \in_a i$, contradicción.

Para probar 2) basta ver que si $c_i \in c_j$ entonces $c_i = c_k$, para cierto $k \in_a j$, luego $i = k$ por 1), luego $i \in_a j$. ∎

[24] La fórmula es Δ_1.

Teorema 5.73 *$x \subset \omega \wedge x$ finito $\to \bigvee n \in \omega \bigwedge i \in \omega(i \in_a n \leftrightarrow i \in x)$.*

DEMOSTRACIÓN: Fijemos $f : m \longrightarrow x$ biyectiva y sea $k = m + 1$. Vamos a probar por inducción sobre u la fórmula

$$u \in k \to \bigvee n(n \in \omega \wedge \bigwedge i \in \omega(i \in_a n \leftrightarrow \bigvee j \in u\, i = f(j))).$$

No es inmediato que esta fórmula sea de tipo Σ_1, pero es equivalente a

$$u \in k \to \bigvee n(n \in \omega \wedge \bigwedge j \in u \bigvee i \in x(i = f(j) \wedge i \in_a n) \wedge$$

$$\bigwedge i \in n(i \in_a n \to \bigvee j \in u\, i = f(j))),$$

que ciertamente es Σ_1. Para probar la equivalencia tenemos en cuenta que en $\mathrm{I}\Sigma_1$ se prueba $\bigwedge in(i \in n \to i < n)$, luego en KP se demuestra la traducción $\bigwedge in \in \omega(i \in_a n \to i \in n)$ (porque en 5.43 hemos visto que $<$ se traduce por $\in$). En particular, en la fórmula precedente vemos que $i \in_a n$ ya implica $i \in n$, por lo que se puede suprimir el $i \in n$ y así es fácil pasar a la fórmula inicial.

Para $u = 0$ basta tomar $n = 0$. Si suponemos que la fórmula para u y se cumple que $u + 1 \in k$, entonces, por hipótesis de inducción tenemos un $n' \in \omega$ tal que

$$\bigwedge i \in \omega(i \in_a n \leftrightarrow \bigvee j \in u\, i = f(j))$$

Ahora usamos que en $\mathrm{I}\Sigma_1$ se demuestra que

$$\bigwedge xi \bigvee y \bigwedge u(u \in y \leftrightarrow u \in x \vee u = i),$$

luego en KP tenemos la traducción

$$\bigwedge x \in \omega \bigwedge i \in \omega \bigvee y \in \omega \bigwedge u \in \omega(u \in_a y \leftrightarrow u \in_a x \vee u = i).$$

En particular, existe $n \in \omega$ tal que

$$\bigwedge i \in \omega(i \in_a n \leftrightarrow i \in_a n' \vee i = f(u)).$$

Al combinar esto con la hipótesis de inducción resulta

$$\bigwedge i \in \omega(i \in_a n \leftrightarrow \bigvee j \in u + 1\, i = f(j)).$$

Esto termina la inducción y, aplicando lo que hemos probado a $u = m$ obtenemos el enunciado. ∎

Teorema 5.74 *Son equivalentes:*

1. *Todo conjunto es finito.*
2. *$\bigwedge x \bigvee i \in \omega\ x = c_i$.*

DEMOSTRACIÓN: Cada c_i es finito, pues $f = \{(i,x) \in n \times c_n \mid x = c_i\}$ es una biyección entre $\{i \in n \mid i \in_a n\}$ y c_n. Esto nos da una implicación. Supongamos ahora que todo conjunto es finito pero $\bigvee x \bigwedge i \in \omega\ x \neq c_i$. Por Π_1-regularidad (aplicada a la fórmula $\bigwedge i \in \omega\, x \neq c_i$) existe un x tal que $\bigwedge i \in \omega\ x \neq c_i$ y $\bigwedge y \in x \bigvee i \in \omega\ y = c_i$.

Aplicando Σ_1-reemplazo a la fórmula $\phi(y,i) \equiv y = c_i$ (teniendo en cuenta la unicidad dada por el teorema anterior) obtenemos un conjunto $\bar{x} \subset \omega$ y una biyección $f : \bar{x} \longrightarrow x$ tal que $\bigwedge i \in \bar{x}\, f(i) = c_i$. En este punto usamos 1), que nos asegura que $\bar{x}$ es finito. Esto nos permite aplicar el teorema anterior, que nos da un $n \in \omega$ tal que $\bigwedge i \in \omega(i \in_a n \leftrightarrow i \in \bar{x})$, lo que a su vez equivale a $\bigwedge i \in \omega(i \in_a n \leftrightarrow c_i \in x)$. Entonces $x = c_n$, pues si $u \in x$, entonces existe un i tal que $u = c_i$, con lo que $c_i \in x$, luego $i \in_a n$, luego $u = c_i \in c_n$, e igualmente se razona la implicación opuesta. Pero habíamos tomado x de modo que fuera distinto de todo c_n, y así tenemos una contradicción. ∎

Por consiguiente, en $\mathrm{KP}_{\mathrm{fin}}$ tenemos que el término

$$\bar{x} \equiv i \mid (i \in \omega \wedge x = c_i)$$

es una descripción propia y es Δ_1, pues $\bar{x} = i$ equivale a $x = c_i$.

Teorema 5.75 *Para toda fórmula $\phi(x_1,\ldots,x_n)$ de $\mathcal{L}_{\mathrm{tc}}$ con las variables libres entre las indicadas, se cumple*

$$\underset{\mathrm{KP}_{\mathrm{fin}}}{\vdash} \bigwedge x_1 \cdots x_n(\phi(x_1,\ldots,x_n) \leftrightarrow (\phi_a)_{\mathrm{tc}}(\bar{x}_1,\ldots,\bar{x}_n)).$$

DEMOSTRACIÓN: Veamos, por inducción[25] sobre la longitud de una expresión θ, que si θ es una fórmula se cumple el enunciado y si es un término se cumple

$$\underset{\mathrm{KP}_{\mathrm{fin}}}{\vdash} \bigwedge x_1 \cdots x_n(\overline{\theta(x_1,\ldots,x_n)} = (\theta_a)_{\mathrm{tc}}(\bar{x}_1,\ldots,\bar{x}_n)).$$

Si $\theta \equiv x$ es trivial, pues $(\theta_a)_{\mathrm{tc}} \equiv x$.

Si $\theta \equiv t_1 \in t_2$, entonces $(\theta_a)_{\mathrm{tc}} \equiv ((t_1)_a)_{\mathrm{t_c}} \in_a ((t_2)_a)_{\mathrm{t_c}}$. Por lo tanto

$$((t_1)_a)_{\mathrm{t_c}}(\bar{x}_1,\ldots,\bar{x}_n) \in_a ((t_2)_a)_{\mathrm{t_c}}(\bar{x}_1,\ldots,\bar{x}_n)$$

equivale a

$$\overline{t_1(x_1,\ldots,x_n)} \in_a \overline{t_2(x_1,\ldots,x_n)}.$$

Si llamamos i y j a ambos miembros, tenemos que $i \in_a j$, luego $c_i \in c_j$, es decir, $t_1(x_1,\ldots,x_n) \in t_2(x_1,\ldots,x_n)$, y esto es θ.

El caso $\theta \equiv t_1 = t_2$ es análogo. Si vale para ϕ y ψ es inmediato que vale para $\neg\phi$ y $\phi \vee \psi$. Supongamos que $\theta \equiv \bigwedge x \phi(x,x_1,\ldots,x_n)$. Por hipótesis de inducción

$$\bigwedge x x_1 \cdots x_n(\phi(x,x_1,\ldots,x_n) \leftrightarrow (\phi_a)_{\mathrm{tc}}(\bar{x},\bar{x}_1,\ldots,\bar{x}_n)),$$

pero esto implica

$$\bigwedge x_1 \cdots x_n(\bigwedge x\, \phi(x,x_1,\ldots,x_n) \leftrightarrow \bigwedge x\, (\phi_a)_{\mathrm{tc}}(\bar{x},\bar{x}_1,\ldots,\bar{x}_n)).$$

[25] Véase la nota en la demostración del teorema 5.47.

Ahora usamos que el hecho de que x recorra todos los conjuntos equivale a que $\bar{x}$ recorra todos los números naturales, luego la fórmula anterior equivale a

$$\bigwedge x_1 \cdots x_n(\theta(x_1,\ldots,x_n) \leftrightarrow \bigwedge x \in \omega\, (\phi_a)_{\mathrm{tc}}(x,\bar{x}_1,\ldots,\bar{x}_n)),$$

que a su vez equivale a

$$\bigwedge x_1 \cdots x_n(\theta(x_1,\ldots,x_n) \leftrightarrow ((\bigwedge x\, \phi)_a)_{\mathrm{tc}}(\bar{x}_1,\ldots,\bar{x}_n)),$$

y la última fórmula es $(\theta_a)_{\mathrm{tc}}$.

El caso de $\bigvee x\phi$ se trata análogamente. Por último, si $\theta \equiv x|\phi(x,x_1,\ldots,x_n)$, entonces $(\theta_a)_{\mathrm{tc}} \equiv x|(x \in \omega \wedge (\phi_a)_{\mathrm{tc}})$ y, con la misma hipótesis de inducción del caso anterior, se concluye claramente que

$$\bigwedge x_1 \cdots x_n(\overset{1}{\bigvee} x\, \phi(x,x_1,\ldots,x_n) \leftrightarrow \overset{1}{\bigvee} x(\phi_a)_{\mathrm{tc}}(\bar{x},\bar{x}_1,\ldots,\bar{x}_n)).$$

Es claro que esto equivale a

$$\bigwedge x_1 \cdots x_n(\overset{1}{\bigvee} x\, \phi(x,x_1,\ldots,x_n) \leftrightarrow \overset{1}{\bigvee} x \in \omega(\phi_a)_{\mathrm{tc}}(x,\bar{x}_1,\ldots,\bar{x}_n)).$$

Dados $x_1,\ldots,x_n$, si se dan las dos partes de la equivalencia y $x = x|\phi$, entonces $\phi(x,x_1,\ldots,x_n)$, luego por hipótesis de inducción tenemos también que $\bar{x} \in \omega \wedge (\phi_a)_{\mathrm{tc}}(\bar{x},\bar{x}_1,\ldots,\bar{x}_n)$, luego $\bar{x} = x|(x \in \omega \wedge (\phi_a)_{\mathrm{tc}})$, es decir, se cumple que $\bar{\theta} = (\theta_a)_{\mathrm{tc}}$, como había que probar.

Si no se dan las unicidades, entonces $\bar{\theta} = (\theta_a)_{\mathrm{tc}}$, pues ambas descripciones son impropias. ∎

En particular, si γ es una sentencia de $\mathcal{L}_{\mathrm{tc}}$, tenemos que

$$\underset{\mathrm{KP}_{\mathrm{fin}}}{\vdash} (\gamma \leftrightarrow (\gamma_a)_{\mathrm{tc}}).$$

Teorema 5.76 *Si γ es una sentencia de $\mathcal{L}_{\mathrm{tc}}$, entonces $\underset{\mathrm{KP}_{\mathrm{fin}}}{\vdash} \gamma$ si y sólo si $\underset{\mathrm{I}\Sigma_1}{\vdash} \gamma_a$.*

Demostración: Si $\underset{\mathrm{KP}_{\mathrm{fin}}}{\vdash} \gamma$, entonces $\underset{\mathrm{I}\Sigma_1}{\vdash} \gamma_a$ porque $\mathrm{I}\Sigma_1$ interpreta a $\mathrm{KP}_{\mathrm{fin}}$ (teorema 5.70). Recíprocamente, si $\underset{\mathrm{I}\Sigma_1}{\vdash} \gamma_a$, entonces $\underset{\mathrm{KP}}{\vdash} (\gamma_a)_{\mathrm{tc}}$ porque KP interpreta a $\mathrm{I}\Sigma_1$ (teorema 5.45), luego $\underset{\mathrm{KP}_{\mathrm{fin}}}{\vdash} \gamma$ por la observación previa al teorema. ∎

Así pues, los teoremas conjuntistas demostrables en $\mathrm{I}\Sigma_1$ son exactamente las traducciones de teoremas de $\mathrm{KP}_{\mathrm{fin}}$. Podemos pensar que KP es la teoría que resulta de extirpar la finitud a $\mathrm{I}\Sigma_1$, de modo que, al contrario que $\mathrm{I}\Sigma_1$, la teoría KP es susceptible de ser extendida con un axioma que postule la existencia de conjuntos infinitos.

También podemos caracterizar los teoremas conjuntistas de la aritmética de Peano:

Definición 5.77 Llamamos $\mathrm{ZFC}_{\mathrm{fin}}$ a la teoría de conjuntos ZFC sin el axioma de infinitud, más el axioma que afirma que todo conjunto es finito, más el axioma CT (clausura transitiva)

$$\bigwedge x \bigvee y(x \subset y \wedge \bigwedge u \in y\, u \subset y),$$

que afirma que todo conjunto está contenido en un conjunto transitivo.

Teorema 5.78 AP *interpreta a* $\mathrm{ZFC}_{\mathrm{fin}}$*, que extiende a* $\mathrm{KP}_{\mathrm{fin}}$.

Demostración: Por 5.50 sabemos que AP interpreta a ZFC sin el axioma de infinitud. Por 5.70 sabemos que $\mathrm{I}\Sigma_1$ (luego AP) interpreta el axioma de finitud, y el teorema 2.18 muestra que ARP, luego $\mathrm{I}\Sigma_1$, prueba la traducción del axioma CT.

Para la segunda parte del enunciado obseervamos que los axiomas de extensionalidad, par y unión de KP son teoremas de $\mathrm{ZFC}_{\mathrm{fin}}$ porque son axiomas, y el esquema de Δ_0-especificación está contenido en el esquema de especificación de Z.

Podemos probar el esquema de regularidad para fórmulas ϕ arbitrarias (no necesariamente Π_1). En efecto, si suponemos que existe un u_0 que cumple $\phi(u_0)$, tomamos un conjunto transitivo x tal que $u_0 \subset x$ y llamamos

$$y = \{u \in x \cup \{u_0\} \mid \phi(u_0)\},$$

que es un conjunto no vacío. El axioma de regularidad de ZF$-$AI afirma que existe un $u \in y$ tal que $u \cap y = \varnothing$. Entonces se cumple $\phi(u)$ y si un $v \in u$ cumpliera $\phi(v)$, tendríamos que $v \in u \in x \cup \{u_0\}$. Si $u \in x$, entonces $v \in x$ porque x es transitivo, y si $u = u_0$, entonces $v \in u_0 \subset x$, luego en cualquier caso $v \in x$, luego $v \in u \cap y$, contradicción.

Por último probamos el axioma de Δ_0-recolección (aunque lo probamos, de hecho, para fórmulas arbitrarias). Suponemos $\bigwedge u \bigvee v \phi(u, v)$, y vamos a probar por inducción sobre n que

$$\bigwedge a(|a| = n \to \bigvee b \bigwedge u \in a \bigvee v \in b\, \phi(u, v)).$$

Si $|a| = 0$ basta tomar $b = \varnothing$. Si vale para n y $|a| = n + 1$, tomamos $u_0 \in a$, de modo que $a = a' \cup \{u_0\}$, donde $|a'| = n$. Por hipótesis de inducción existe b' tal que $\bigwedge u \in a' \bigvee v \in b'\, \phi(u, v)$, y podemos tomar v_0 tal que $\phi(u_0, v_0)$, con lo que $b \cup \{v_0\}$ cumple lo requerido. ∎

Ahora la prueba del teorema 5.76 vale sin cambio alguno para probar:

Teorema 5.79 *Si* γ *es una sentencia de* $\mathcal{L}_{\mathrm{tc}}$*, entonces* $\underset{\mathrm{ZFC}_{\mathrm{fin}}}{\vdash} \gamma$ *si y sólo si* $\underset{\mathrm{AP}}{\vdash} \gamma_a$.

De hecho, los axiomas de partes, reemplazo o elección no han hecho falta para probar el teorema 5.78, por lo que este teorema vale igualmente eliminando estos axiomas. Esto significa que todos ellos son demostrables en $\mathrm{ZFC}_{\mathrm{fin}}$.

Vemos así que en AP se pueden demostrar exactamente las traducciones de los teoremas de $\mathrm{ZFC}_{\mathrm{fin}}$.

Capítulo VI

Teorías y metateorías

En el capítulo I introdujimos (la versión original de) la aritmética recursiva primitiva, y lo hicimos razonando informalmente: hablábamos de signos (que pueden ser cualquier cosa, desde formas de trazos que podemos escribir en un papel hasta números naturales, pasando por conceptos abstractos como "alfil", "torre", etc. en el juego del ajedrez), y seleccionamos algunas cadenas de signos a las que llamar fórmulas, les asignamos un significado, y fijamos unos criterios puramente formales para determinar cuándo una sucesión de fórmulas constituye una demostración en ARP, justificando, no obstante, que todos los teoremas de ARP son necesariamente verdaderos.

Del mismo modo que hemos construido y estudiado ARP informalmente, podríamos construir y estudiar informalmente cualquier otra de las teorías axiomáticas de primer orden que hemos estudiado, como la aritmética de Peano AP. Igual que hemos tomado ciertos signos como signos básicos de $\mathcal{L}_{\text{arp}}$, podemos elegir signos cualesquiera para referirnos a los conceptos primitivos de AP, y seleccionar las cadenas de signos que constituyen fórmulas, y definir las que tomamos como axiomas, fijar unas reglas de inferencia y estudiar qué se puede demostrar a partir de esos axiomas y esas reglas, y su relación con otras teorías similares.

Ahora bien, no es eso exactamente lo que hemos hecho en el capítulo IV. En lugar de estudiar AP informalmente, razonando en castellano sobre las propiedades de unos conceptos definidos con toda precisión (los signos de $\mathcal{L}_a$, las fórmulas de $\mathcal{L}_a$, los axiomas de AP, los teoremas de AP, etc.) hemos expresado estos hechos en el lenguaje $\mathcal{L}_{\text{arp}}$ y los hemos demostrado razonando formalmente en ARP. Hemos visto que cada fórmula de ARP tiene una interpretación precisa como afirmación sobre los números naturales y las funciones recursivas primitivas, y todos los teoremas de ARP son verdaderos. Con el "truco" de identificar los signos de $\mathcal{L}_a$ con números naturales, algunas afirmaciones de $\mathcal{L}_{\text{arp}}$ se interpretan como afirmaciones sobre las fórmulas, los axiomas, las reglas de inferencia y los teoremas de AP, de modo que es indistinto convencerse de que un hecho sobre AP es cierto razonándolo informalmente en castellano como razonándolo formalmente en ARP. La ventaja de razonar formalmente en ARP,

como hemos hecho, es que así sabemos exactamente qué principios básicos son necesarios para llegar a las conclusiones a las que hemos llegado (los expresados por los axiomas y reglas de inferencia de ARP), que en este caso son principios finitistas en el sentido más restrictivo del término.

Cuando estudiamos una teoría axiomática (como AP) razonando formalmente en otra teoría axiomática (como ARP), se dice que la segunda es la *metateoría* que usamos para estudiar la teoría de la que hablamos. Así, hemos visto que ARP puede usarse como metateoría para estudiar la propia ARP y cualquier otra teoría axiomática de primer orden. En este capítulo vamos a estudiar con más detalle las relaciones que se dan entre una teoría formal usada como metateoría y otra teoría formal estudiada en ella. Esto nos lleva a trabajar a tres niveles que tendremos que distinguir cuidadosamente para que los enunciados que probemos no resulten confusos:

1. En principio, tomaremos como metateoría una versión informal de $\mathrm{I}\Sigma_1$ (ya hemos visto que no hay razón para trabajar en ARP cuando podemos trabajar en $\mathrm{I}\Sigma_1$), de modo que consideraremos un lenguaje formal $\mathcal{L}_a^0$ formado por signos $0, S, +$, etc. "de verdad", "de los que se pueden escribir en un papel", de modo que todas las afirmaciones que haremos en este capítulo pueden verse como descripciones informales de fórmulas de $\mathcal{L}_a^0$ demostrables formalmente en $\mathrm{I}\Sigma_1$.

   Decimos "en principio" porque hemos demostrado (teorema 4.49) que si extendemos $\mathcal{L}_a^0$ hasta el lenguaje de la aritmética recursiva primitiva (añadiendo funtores) y tomamos como axiomas las definiciones de las funciones recursiva primitiva obtenemos una extensión intrascendente, de modo que toda sentencia de $\mathcal{L}_a^0$ demostrable en la extensión lo es también en $\mathrm{I}\Sigma_1$. Por lo tanto, podemos considerar esta extensión siempre que lo consideremos conveniente, aunque también hemos visto que los funtores de $\mathcal{L}_{\mathrm{arp}}$ pueden sustituirse por descripciones adecuadas en $\mathcal{L}_a$. En general será preferible considerar el lenguaje básico $\mathcal{L}_a^0$ para mantenerlos lo más cerca posible del segundo nivel:

2. En la metateoría $\mathrm{I}\Sigma_1$ podemos definir el lenguaje formal $\mathcal{L}_a$ (que no debemos confundir con $\mathcal{L}_a^0$), y sobre él estudiaremos formalmente varias teorías axiomáticas, entre ellas las teorías $\mathrm{I}\Sigma_n$ y la aritmética de Peano AP.

   En 4.9 definimos $\mathcal{L}_a$ sin precisar qué número concreto es cada uno de sus signos, porque eso resulta irrelevante a todos los efectos, pero para poner ejemplos ilustrativos vamos a concretar la definición mediante la tabla siguiente:

$\mathcal{L}_a^0$	0	S	$+$	$\cdot$	$=$	$\neg$	$\vee$	$\bigwedge$	$\bigvee$	$\mid$	x_i	u_i
$\mathcal{L}_a$	$\mathbf{0}$	$\mathbf{1}$	$\mathbf{2}$	$\mathbf{3}$	$\mathbf{4}$	$\mathbf{5}$	$\mathbf{6}$	$\mathbf{7}$	$\mathbf{8}$	$\mathbf{9}$	$\mathbf{2}^{i+\mathbf{4}}$	$\mathbf{3}^{i+\mathbf{3}}$

   En esta tabla hemos adoptado el convenio de representar en negrita los numerales de $\mathcal{L}_a^0$, de modo que, por ejemplo,

$$\mathbf{5} = SSSSS\mathbf{0} = \mathbf{0}'''''.$$

Esta cadena de signos "que puede escribirse en un papel" es, por definición, el negador $\neg$ de $\mathcal{L}_a$, de modo que sería un sinsentido escribir, por ejemplo, $\neg = \neg$ en $\mathcal{L}_a^0$ si entendemos que $\neg$ es el negador de $\mathcal{L}_a^0$ y $=$ es su igualador (sería una cadena de signos de $\mathcal{L}_a^0$, pero no una expresión), pero eso mismo tiene perfecto sentido si entendemos, que, según la tabla, $\neg \equiv \mathbf{5}$ es el negador de $\mathcal{L}_a$, mientras que $=$ sigue siendo el igualador de $\mathcal{L}_a^0$. Entonces $\neg = \neg$ es la sentencia $\mathbf{5} = \mathbf{5}$ de $\mathcal{L}_a^0$.

Podríamos evitar estos equívocos llamando $\ulcorner\neg\urcorner = \mathbf{5}$ al negador de $\mathcal{L}_a$ para distinguirlo del de $\mathcal{L}_a^0$, pero no será necesario porque no vamos a tener ocasión de combinar los signos de $\mathcal{L}_a^0$ con los de $\mathcal{L}_a$, ya que la metateoría la expresaremos siempre informalmente, como hasta ahora.

Notemos que $\mathcal{L}_a$ tiene sus propios numerales, que no coinciden con los de $\mathcal{L}_a^0$. Por ejemplo,

$$5 = 0'''' = \langle S, S, S, S, S, 0\rangle = \langle \mathbf{1}, \mathbf{1}, \mathbf{1}, \mathbf{1}, \mathbf{1}, \mathbf{0}\rangle = \mathbf{64\,440\,289\,805\,671\,411}.$$

Así, mientras el numeral $\mathbf{5}$ de $\mathcal{L}_a^0$ es una cadena de signos en sentido literal, el numeral 5 de $\mathcal{L}_a$ es uno de los números naturales de los que podemos hablar formalmente en la metateoría $\mathrm{I}\Sigma_1$.

3. En este capítulo estudiaremos más detalladamente la relación entre una teoría y su metateoría, pero no entre la metateoría $\mathrm{I}\Sigma_1$ (sobre $\mathcal{L}_a^0$) y las teorías que definiremos en ella, pues ello nos obligaría a razonar informalmente, es decir, a probar informalmente resultados sobre la metateoría. En su lugar, en la teoría $\mathrm{I}\Sigma_1$ sobre $\mathcal{L}_a$ definiremos el lenguaje formal $\ulcorner\mathcal{L}_a\urcorner$, sobre el cual consideraremos las teorías $\ulcorner\mathrm{I}\Sigma_n\urcorner$, $\ulcorner\mathrm{AP}\urcorner$, etc. y así, razonando formalmente en la metateoría $\mathrm{I}\Sigma_1$ sobre $\mathcal{L}_a^0$, estudiaremos la relación entre la teoría $\mathrm{I}\Sigma_1$ y las teorías formalizadas en ella.

   La definición de $\ulcorner\mathcal{L}_a\urcorner$ es simplemente la formalización en la teoría $\mathrm{I}\Sigma_1$ de la definición de $\mathcal{L}_a$ en la metateoría, de modo que tenemos la correspondencia que muestra la tabla siguiente:

$\mathcal{L}_a^0$	0	S	$+$	$\cdot$	$=$	$\neg$	$\vee$	$\bigwedge$	$\bigvee$	$\mid$	x_i	u_i
$\mathcal{L}_a$	$\mathbf{0}$	$\mathbf{1}$	$\mathbf{2}$	$\mathbf{3}$	$\mathbf{4}$	$\mathbf{5}$	$\mathbf{6}$	$\mathbf{7}$	$\mathbf{8}$	$\mathbf{9}$	$\mathbf{2}^{i+4}$	$\mathbf{3}^{i+3}$
$\ulcorner\mathcal{L}_a\urcorner$	0	1	2	3	4	5	6	7	8	9	2^{i+4}	3^{i+3}

   Ahora sí que convendrá a menudo (cuando sirva para evitar confusiones, pero no cuando sólo vuelva farragosa la notación) escribir $\ulcorner\neg\urcorner$ en lugar de $\neg$ para referirnos al negador de $\ulcorner\mathcal{L}_a\urcorner$ (el numeral 5 de $\mathcal{L}_a$) y distinguirlo así del negador de $\mathcal{L}_a$ (el numeral $\mathbf{5}$ de $\mathcal{L}_a^0$).

Luego incidiremos más a fondo en las relaciones sutiles que se dan entre estos tres niveles de lenguaje (especialmente entre el segundo y el tercero), pero de momento dedicamos la primera sección a estudiar una teoría aritmética especialmente simple.

6.1 La aritmética de Robinson

Definición 6.1 Llamaremos *aritmética de Robinson* a la teoría axiomática Q sobre el lenguaje $\mathcal{L}_a$ (sin el relator $\leq$) cuyos axiomas son:

$$\begin{array}{ll}
(\mathrm{Q1}) & x' \neq 0 \\
(\mathrm{Q2}) & x' = y' \to x = y \\
(\mathrm{Q3}) & x \neq 0 \to \bigvee u\ x = u' \\
(\mathrm{Q4}) & x + 0 = x \\
(\mathrm{Q5}) & x + y' = (x + y)' \\
(\mathrm{Q6}) & x \cdot 0 = 0 \\
(\mathrm{Q7}) & x \cdot y' = xy + x
\end{array}$$

Equivalentemente, es la teoría que resulta de eliminar el principio de inducción de la aritmética IA con inducción abierta. Podría parecer una teoría demasiado débil para que pueda probarse en ella nada relevante, pero veremos que no es así, que podemos probar hechos no triviales sobre ella y el hecho de que sea una teoría tan simple hace que estos resultados se apliquen a prácticamente cualquier teoría aritmética.

Como el interés de Q es puramente teórico, vamos a considerarla únicamente sobre el lenguaje formal $\mathcal{L}_a$ sin descriptor.

En los teoremas siguientes será fundamental distinguir los números naturales de metamatemáticos (los de $\mathcal{L}_a^0$) de los numerales de $\mathcal{L}_a$. Para ello introducimos el término $0^{(n)}$, determinado por los teoremas

$$0^{(\mathbf{0})} = 0, \qquad 0^{(n+1)} = (0^{(n)})',$$

de modo que

$$0^{(\mathbf{0})} = 0, \qquad 0^{(\mathbf{1})} = 0' = 1, \qquad 0^{(\mathbf{2})} = 0'' = 2, \qquad \ldots$$

son los numerales de $\mathcal{L}_a$. Más detalladamente:

$$0^{(\mathbf{2})} = \langle S, S, 0\rangle = \langle \mathbf{1}, \mathbf{1}, \mathbf{0}\rangle = \mathbf{139}.$$

Observemos que $0^{(n)}$ es un término (metamatemático) con n como única variable libre, pero dicho término hace referencia a un designador de $\mathcal{L}_a$ sin variables libres. Al variar n varía el designador, pero $0^{(n)}$ siempre es un designador. Consta n funtores sucesor y un 0, pero sin ninguna variable.

Recordemos que la relación de orden se define como[1]

$$x \leq y \equiv \bigvee u\ u + x = y.$$

Veamos un primer teorema básico sobre Q:

[1] En Q no puede probarse que $x + y = y + x$, por lo que no sería equivalente definir $x \leq y \equiv \bigvee u\, x + u = y$.

Teorema 6.2 *Se cumple:*

1. $\bigwedge mn \underset{Q}{\vdash} 0^{(m+n)} = 0^{(m)} + 0^{(n)}$.
2. $\bigwedge mn \underset{Q}{\vdash} 0^{(mn)} = 0^{(m)} \cdot 0^{(n)}$.
3. $\bigwedge mn(m \neq n \to \underset{Q}{\vdash} 0^{(m)} \neq 0^{(n)})$.
4. $\bigwedge mn(m \leq n \to \underset{Q}{\vdash} 0^{(m)} \leq 0^{(n)})$.
5. $\bigwedge mn(n < m \to \underset{Q}{\vdash} \neg 0^{(m)} \leq 0^{(n)})$.

Demostración: 1) Fijado m, razonamos por inducción[2] sobre n. Esto es correcto, pues la fórmula $\phi(n) \equiv \underset{Q}{\vdash} 0^{(m+n)} = 0^{(m)} + 0^{(n)}$ es ciertamente Σ_1.

Para $n = 0$ hay que probar $\underset{Q}{\vdash} 0^{(m)} = 0^{(m)} + 0$, y esto es consecuencia lógica de Q4. Si suponemos el resultado para n, la demostración en Q puede prolongarse aplicando S a los dos miembros, con lo que

$$\underset{Q}{\vdash} (0^{(m+n)})' = (0^{(m)} + 0^{(n)})'.$$

Por la definición de los numerales $(0^{(m+n)})' \equiv 0^{(m+n+1)}$, y aplicando Q5 y de nuevo la definición de los numerales obtenemos

$$\underset{Q}{\vdash} 0^{(m+(n+1))} = 0^{(m)} + 0^{(n+1)}.$$

La prueba de 2) es análoga. Para probar 3) suponemos $m < n$ y razonamos por inducción sobre k que

$$k \leq m \to \underset{Q}{\vdash} 0^{(m)} = 0^{(n)} \to 0^{(m \dot{-} k)} = 0^{(n \dot{-} k)}.$$

En efecto, para $k = 0$ es inmediato y, si vale para k y $k + 1 \leq m$, tenemos que $0^{(m \dot{-} k)} \equiv S\, 0^{(m \dot{-} (k+1))}$, $0^{(n \dot{-} k)} \equiv S\, 0^{(n \dot{-} (k+1))}$, y aplicando Q2 obtenemos

$$\underset{Q}{\vdash} 0^{(m \dot{-} k)} = 0^{(n \dot{-} k)} \to 0^{(m \dot{-} (k+1))} = 0^{(n \dot{-} (k+1))},$$

de donde se sigue la conclusión para $k + 1$. Aplicando el resultado a $k = m$ obtenemos que

$$\underset{Q}{\vdash} 0^{(m)} = 0^{(n)} \to 0 = 0^{(n \dot{-} m)},$$

pero $0^{(n \dot{-} m)} \equiv (0^{(n \dot{-} (m+1))})'$ y podemos concluir $0^{(m)} \neq 0^{(n)}$ aplicando Q1.

[2]Notemos que estamos probando por inducción sobre n en la metateoría $\mathrm{I}\Sigma_1$ que cada sentencia es demostrable en Q (sin usar inducción).

Para probar 4) observamos que, por 1),

$$\underset{Q}{\vdash} 0^{(n \dot{-} m)} + 0^{(m)} = 0^{(n)},$$

de donde deducimos $0^{(m)} \leq 0^{(n)}$.

Para probar 5), si $n < m$, razonamos por inducción sobre k que

$$k \leq n \to \underset{Q}{\vdash} x + 0^{(m)} = 0^{(n)} \to x + 0^{(m \dot{-} k)} = 0^{(n \dot{-} k)},$$

de donde en particular

$$\underset{Q}{\vdash} x + 0^{(m)} = 0^{(n)} \to x + 0^{(m \dot{-} n)} = 0,$$

pero $0^{(m \dot{-} n)} \equiv (0^{(m \dot{-} (n+1))})'$, luego aplicando Q1 podemos concluir que

$$\underset{Q}{\vdash} x + 0^{(m)} \neq 0^{(n)}.$$

Introduciendo el generalizador y aplicando la regla de negación del particularizador obtenemos

$$\underset{Q}{\vdash} \neg \bigvee u \, u + 0^{(m)} = 0^{(n)},$$

que es lo mismo que $\underset{Q}{\vdash} \neg 0^{(m)} \leq 0^{(n)}$. ∎

Todas las fórmulas del teorema anterior eran sentencias. Veamos que en Q también es posible demostrar algunos resultados generales.

Teorema 6.3 *Se cumple:*

1. $\underset{Q}{\vdash} x + y = 0 \to x = 0 \wedge y = 0.$
2. $\underset{Q}{\vdash} xy = 0 \to x = 0 \vee y = 0.$
3. $\underset{Q}{\vdash} 0 \leq x.$
4. $\bigwedge n \ \underset{Q}{\vdash} x \leq 0^{(n+\mathbf{1})} \to x \leq 0^{(n)} \vee x = 0^{(n+\mathbf{1})}.$
5. $\bigwedge n \ \underset{Q}{\vdash} x + 1 \leq 0^{(n+\mathbf{1})} \to x \leq 0^{(n)}.$
6. $\bigwedge n \ \underset{Q}{\vdash} (x + 1) + 0^{(n)} = x + 0^{(n+\mathbf{1})}.$

Demostración: Los teoremas 1) y 2) están probados en 4.10, en principio en IA, pero las pruebas no usan el principio de inducción, luego valen en Q. Lo mismo sucede con 3), aunque la prueba es inmediata, pues se sigue de que $x + 0 = x$.

Para probar 4) razonamos por inducción sobre n. Para $n = 0$ hay que probar en Q que

$$x \leq 0^{(\mathbf{1})} \to x \leq 0^{(\mathbf{0})} \vee x = 0^{(\mathbf{1})}.$$

Si $x \leq 0^{(\mathbf{1})}$, existe un y tal que $y + x = 0^{(\mathbf{1})}$. Distinguimos dos casos, si $x = 0$ la conclusión es inmediata, y en caso contrario, por Q3 tenemos que existe un x tal que $x = z + 0^{(\mathbf{1})}$, con lo que tenemos que $y + (z + 1) = 0^{(1)}$, que equivale a $(y + z)' = (0^{(\mathbf{0})})'$, luego $y + z = 0^{(\mathbf{0})}$, luego $z = 0$, luego $x = 0^{(\mathbf{1})}$.

Supuesto cierto para n, suponemos que $x \leq 0^{(n+\mathbf{2})}$, de modo que existe un y tal que $y + x = 0^{(n+\mathbf{2})}$. Si $x = 0$ la conclusión es inmediata y, en caso contrario, existe un x tal que $x = z'$, con lo que $y + z' = 0^{(n+\mathbf{2})}$, que equivale a $(y+z)' = (0^{(n+\mathbf{1})})'$, de donde $y+z = 0^{(n+\mathbf{1})}$, es decir, $z \leq 0^{(n+\mathbf{1})}$. Por hipótesis de inducción, $z \leq 0^{(n)} \vee z = 0^{(n+\mathbf{1})}$. En el primer caso existe un y tal que $y + z = 0^{(n)}$, de donde $y + x = 0^{(n+\mathbf{1})}$, luego $x \leq 0^{(n+\mathbf{1})}$, y en el segundo caso $x = 0^{(n+\mathbf{2})}$.

Para probar 5) observamos que la hipótesis nos da $y + x + 1 = 0^{(n)} + 1$, de donde $y + x = 0^{(n)}$, y a su vez $x \leq 0^{(n)}$.

6) Se razona por inducción sobre n. Para $n = 0$ es inmediato y, si vale para n, completamos la demostración aplicando S a ambos miembros y usando el axioma Q5. ∎

Con esto estamos en condiciones de probar lo siguiente:

Teorema 6.4 *Si n es un número natural, las fórmulas siguientes son demostrables en* Q*:*

1. $x \leq 0^{(n)} \leftrightarrow x = 0^{(\mathbf{0})} \vee \cdots \vee x = 0^{(n)}$.
2. $0^{(n)} \leq x \leftrightarrow 0^{(n)} = x \vee 0^{(n+\mathbf{1})} \leq x$.
3. $x \leq 0^{(n)} \vee 0^{(n)} \leq x$

Demostración: 1) Para probar que

$$\bigwedge n \underset{Q}{\vdash} \bigvee_{i\leq n} x = 0^{(i)} \to x \leq 0^{(n)}$$

razonamos por inducción sobre n que

$$n \leq m \to \underset{Q}{\vdash} \bigvee_{i\leq n} x = 0^{(i)} \to x \leq 0^{(m)}.$$

Para $n = 0$ hay que probar que $\underset{Q}{\vdash} x = 0 \to x \leq 0^{(m)}$, lo cual es cierto.

Si se cumple para n y $n + 1 \leq m$, por hipótesis de inducción tenemos que

$$\underset{Q}{\vdash} \bigvee_{i\leq n} x = 0^{(i)} \to x \leq 0^{(m)},$$

y basta observar que $\underset{Q}{\vdash} x = 0^{(n+1)} \to x \leq 0^{(m)}$, con lo que

$$\underset{Q}{\vdash} \bigvee_{i\leq n} x = 0^{(i)} \vee x = 0^{(n+1)} \to x \leq 0^{(m)}.$$

Esto es lo mismo que $\underset{Q}{\vdash} \bigvee_{i\leq n+1} x = 0^{(i)} \to x \leq 0^{(m)}$. Haciendo $m = n$ obtenemos la conclusión.

Demostramos la implicación contraria por inducción sobre n. Para $n = \mathbf{0}$ se reduce a

$$\underset{Q}{\vdash} x \leq 0 \leftrightarrow x = 0,$$

lo cual se sigue del apartado 1) del teorema anterior, teniendo en cuenta[3] que $x \leq 0 \equiv \bigvee u\ u + x = 0$. Supuesto cierto para n, el apartado 4) del teorema anterior nos da que

$$\underset{Q}{\vdash} x \leq 0^{(n+\mathbf{1})} \to x \leq 0^{(n)} \vee x = 0^{(n+\mathbf{1})}$$

y aplicando la hipótesis de inducción obtenemos inmediatamente la conclusión.

2) Si $0^{(n)} \leq x$, existe un z tal que $z + 0^{(n)} = x$. Si $z = 0$ tenemos que $z + 0^{(n)} = 0^{(n)}$ por 6.2, luego $0^{(n)} = x$. En caso contrario, por el axioma Q3, existe un y tal que $z = y + 0^{(\mathbf{1})}$, luego $x = z + 0^{(n)} = (y + 0^{(\mathbf{1})}) + 0^{(n)}$, y el teorema anterior nos da $y + 0^{(n+\mathbf{1})} = x$, luego $0^{(n+\mathbf{1})} \leq x$.

3) Razonamos por inducción sobre n. Para $n = \mathbf{0}$ se sigue de 6.3.3. Supuesto cierto para n, tenemos $x \leq 0^{(n)} \vee 0^{(n)} \leq x$. En el primer caso $x \leq 0^{(n+\mathbf{1})}$ por el apartado 1. y 6.2.4, luego también $x \leq 0^{(n+\mathbf{1})} \vee 0^{(n+\mathbf{1})} \leq x$.

En el segundo caso, por el apartado 2., $0^{(n)} = x \vee 0^{(n+\mathbf{1})} \leq x$. En el primero de estos dos casos tenemos $x \leq 0^{(n+\mathbf{1})}$, luego en ambos tenemos la conclusión. ∎

Los axiomas de Q son fórmulas abiertas excepto Q3. Vamos a construir una extensión intrascendente de Q cuyos axiomas sean todos fórmulas abiertas.

Definición 6.5 Llamemos $\mathcal{L}_a^*$ al lenguaje formal que resulta de añadir a $\mathcal{L}_a$ un funtor monádico pre, y llamemos Q^* a la teoría axiomática cuyos axiomas son los de Q excepto que el axioma Q3 lo sustituimos por los axiomas:

$$\operatorname{pre} 0 = 0, \qquad x \neq 0 \to x = (\operatorname{pre} x)'.$$

Obviamente el segundo de estos axiomas implica Q3, por lo que todo teorema de Q es también un teorema de Q^*. Vamos a probar que Q^* es una extensión intrascendente de Q.

Para ello observamos en primer lugar que

$$\underset{Q^*}{\vdash} (y = \operatorname{pre} x \leftrightarrow (x = 0 \wedge y = 0) \vee x = y').$$

[3] Si consideramos a $\leq$ como un relator de $\mathcal{L}_a$ la conclusión es inmediata.

En efecto, si suponemos $y = \operatorname{pre} x$, sólo tenemos que distinguir dos casos, según si $x = 0$ o $x \neq 0$. Recíprocamente, si suponemos la fórmula de la derecha, en el primer caso tenemos obviamente que $y = \operatorname{pre} x$, y en el segundo tenemos que $y' = (\operatorname{pre} x)'$, luego por Q2 es $y = \operatorname{pre} x$.

Más en general, a cada término $t(x_1, \ldots, x_n)$ de $\mathcal{L}_a^*$ le asociamos una fórmula $\phi_t(y, x_1, \ldots, x_n)$ (donde y es una variable que no esté en t) de modo que

$$\underset{Q^*}{\vdash} (y = t \leftrightarrow \phi(y, x_1, \ldots, x_n))$$

Si t está en $\mathcal{L}_a$, la equivalencia se demuestra en Q.

La definición es:

1. Si $t \equiv 0$, entonces $\phi_t(y) \equiv y = 0$.
2. Si $t = t'_0$, entonces $\phi_t(y) \equiv \bigvee u(\phi_t(u) \wedge y = u')$.
3. Si $t = \operatorname{pre} t_0$, entonces $\phi_t(y) \equiv \bigvee u(\phi_{t_0}(u) \wedge ((u = 0 \wedge y = 0) \vee u = y'))$.
4. Si $t = t_1 + t_2$, entonces $\phi_t(y) \equiv \bigvee uv(\phi_{t_1}(u) \wedge \phi_{t_2}(v) \wedge y = u + v)$.
5. Si $t = t_1 \cdot t_2$, entonces $\phi_t(y) \equiv \bigvee uv(\phi_{t_1}(u) \wedge \phi_{t_2}(v) \wedge y = uv)$.

Es fácil comprobar que se cumple lo requerido.

A su vez, a cada fórmula α de $\mathcal{L}_a^*$ le asociamos otra fórmula α^* de $\mathcal{L}_a$, con las mismas variables libres, tal que $\underset{Q^*}{\vdash} (\alpha \leftrightarrow \alpha^*)$, y la equivalencia se demuestra en Q si α está en $\mathcal{L}_a$.

Basta definir:

1. Si $\alpha \equiv t_1 = t_2$, entonces $\alpha^* \equiv \bigvee u(\phi_{t_1}(u) \wedge \phi_{t_2}(u))$.
2. Si $\alpha \equiv \neg\beta$, entonces $\alpha^* \equiv \neg\beta^*$.
3. Si $\alpha \equiv \beta \vee \gamma$, entonces $\alpha^* \equiv \beta^* \vee \gamma^*$.
4. Si $\alpha \equiv \bigwedge u\, \beta$, entonces $\alpha^* \equiv \bigwedge u\, \beta^*$.
5. Si $\alpha \equiv \bigvee u\, \beta$, entonces $\alpha^* \equiv \bigvee u\, \beta^*$.

De nuevo es fácil probar que se cumple lo requerido.

Una comprobación rutinaria muestra que si α es un axioma de Q^*, entonces α^* es un teorema de Q. De hecho, sólo hay que comprobarlo para los dos axiomas que sustituyen a Q3, pues los restantes son axiomas de Q y entonces α^* es equivalente a α en Q.

También es inmediato que si α se deduce de otras fórmulas por una de las reglas de inferencia de $K_{\mathcal{L}_a^*}$, entonces α^* se deduce de las traducciones correspondientes por la misma regla. Esto implica que, para toda fórmula α de $\mathcal{L}_a^*$, si $\underset{Q^*}{\vdash} \alpha$, entonces $\underset{Q}{\vdash} \alpha^*$. Así pues:

Teorema 6.6 *Toda fórmula de $\mathcal{L}_a^*$ es equivalente en Q^* a una fórmula de $\mathcal{L}_a$, y una fórmula de $\mathcal{L}_a$ es demostrable en Q^* si y sólo si lo es en* Q.

6.2 $\mathrm{I}\Sigma_1$ y su formalización

Antes de continuar con nuestro estudio de Q en $\mathrm{I}\Sigma_1$ necesitamos demostrar en $\mathrm{I}\Sigma_1$ algunos resultados sobre (la versión formalizada de) el propio $\mathrm{I}\Sigma_1$. En primer lugar observamos que en $\mathrm{I}\Sigma_1$ se demuestra:

$$\mathbf{2}\,\langle x,y\rangle_2 = (x+y)(x+y+\mathbf{1})+\mathbf{2}x,$$

y formalizando la demostración tenemos que en $\mathrm{I}\Sigma_1$ se demuestra

$$\underset{\mathrm{I}\Sigma_1}{\vdash} 0^{(\mathbf{2})}\,\langle x,y\rangle_2 = (x+y)(x+y+0^{(\mathbf{1})})+0^{(\mathbf{2})}x.$$

En particular,

$$\bigwedge mn \underset{\mathrm{I}\Sigma_1}{\vdash} 0^{(\mathbf{2})}\left\langle 0^{(m)},0^{(n)}\right\rangle_2 = (0^{(m)}+0^{(n)})(0^{(m)}+0^{(n)}+0^{(\mathbf{1})})+0^{(\mathbf{2})}0^{(m)},$$

pero, usando el teorema 6.3, de aquí deducimos que

$$\bigwedge mn \underset{\mathrm{I}\Sigma_1}{\vdash} 0^{(\mathbf{2})}\left\langle 0^{(m)},0^{(n)}\right\rangle_2 = 0^{((m+n)(m+n+\mathbf{1})+\mathbf{2}m)},$$

que es lo mismo que

$$\bigwedge mn \underset{\mathrm{I}\Sigma_1}{\vdash} 0^{(\mathbf{2})}\left\langle 0^{(m)},0^{(n)}\right\rangle_2 = 0^{(\mathbf{2}\langle m,n\rangle_2)},$$

o también

$$\bigwedge mn \underset{\mathrm{I}\Sigma_1}{\vdash} 0^{(\mathbf{2})}\left\langle 0^{(m)},0^{(n)}\right\rangle_2 = 0^{(\mathbf{2})}0^{(\langle m,n\rangle_2)},$$

luego

$$\bigwedge mn \underset{\mathrm{I}\Sigma_1}{\vdash} \left\langle 0^{(m)},0^{(n)}\right\rangle_2 = 0^{(\langle m,n\rangle_2)}.$$

Más aún, sabemos que

$$\underset{\mathrm{I}\Sigma_1}{\vdash} z = \langle x,y\rangle_2 \leftrightarrow x = z_1 \wedge y = z_2,$$

luego

$$\bigwedge nab \underset{\mathrm{I}\Sigma_1}{\vdash} 0^{(n)} = \left\langle 0^{(a)},0^{(b)}\right\rangle_2 \leftrightarrow 0^{(a)} = 0_1^{(n)} \wedge 0^{(b)} = 0_2^{(n)},$$

y por el resultado previo, esto equivale a

$$\bigwedge nab \underset{\mathrm{I}\Sigma_1}{\vdash} 0^{(n)} = 0^{(\langle a,b\rangle_2)} \leftrightarrow 0^{(a)} = 0_1^{(n)} \wedge 0^{(b)} = 0_2^{(n)}.$$

Aplicándolo a $a = n_1$ y $b = n_2$, con lo que $\langle a,b\rangle_2 = n$, concluimos que

$$\bigwedge n \underset{\mathrm{I}\Sigma_1}{\vdash} 0_1^{(n)} = 0^{(n_1)} \wedge 0_2^{(n)} = 0^{(n_2)}.$$

Veamos ahora que $\bigwedge n \underset{\mathrm{I}\Sigma_1}{\vdash} 0^{(n)} \dot{-} 1 = 0^{(n \dot{-} \mathbf{1})}$.

Para $n = 0$ se comprueba trivialmente y, si $n \neq 0$, entonces $n \dot{-} \mathbf{1} + \mathbf{1} = n$, luego

$$\underset{\mathrm{I}\Sigma_1}{\vdash} 0^{(n)} \neq 0, \qquad \underset{\mathrm{I}\Sigma_1}{\vdash} 0^{(n \dot{-} \mathbf{1})} + 0^{(\mathbf{1})} = 0^{(n)}.$$

Por otro lado,

$$\underset{\mathrm{I}\Sigma_1}{\vdash} x \neq 0 \to (x \dot{-} 0^{(\mathbf{1})}) + 0^{(\mathbf{1})} = x,$$

luego

$$\underset{\mathrm{I}\Sigma_1}{\vdash} 0^{(n)} \neq 0 \to (0^{(n)} \dot{-} 0^{(\mathbf{1})}) + 0^{(\mathbf{1})} = 0^{(n)},$$

de donde

$$\underset{\mathrm{I}\Sigma_1}{\vdash} 0^{(n \dot{-} \mathbf{1})} + 0^{(\mathbf{1})} = (0^{(n)} \dot{-} \mathbf{1}) + 0^{(\mathbf{1})}$$

y simplificando el $0^{(\mathbf{1})}$ tenemos la conclusión.

De momento, por una mera cuestión estética, en el enunciado del teorema siguiente introducimos la notación $\ulcorner s \urcorner \equiv 0^{(s)}$ para usarla cuando pensemos en s, no como un número natural, sino como en una sucesión finita de números naturales.

Teorema 6.7 $\bigwedge s \underset{\mathrm{I}\Sigma_1}{\vdash} \ell(\ulcorner s \urcorner) = 0^{(\ell(s))}, \qquad \bigwedge i < \ell(s) \underset{\mathrm{I}\Sigma_1}{\vdash} \ulcorner s \urcorner_{0^{(i)}} = 0^{(s_i)}.$

Demostración: El caso $s = \mathbf{0}$ se demuestra aparte y es trivial. Suponemos, pues, que $s \neq \mathbf{0}$, con lo que

$$\underset{\mathrm{I}\Sigma_1}{\vdash} \ulcorner s \urcorner \neq 0^{(\mathbf{0})}, \qquad \underset{\mathrm{I}\Sigma_1}{\vdash} x \neq 0^{(\mathbf{0})} \to \ell(x) = (x \dot{-} 0^{(\mathbf{1})})_1.$$

Por consiguiente, $\underset{\mathrm{I}\Sigma_1}{\vdash} \ell(\ulcorner s \urcorner) = (0^{(s)} \dot{-} 0^{(\mathbf{1})})_1$ y, por los teoremas precedentes,

$$\underset{\mathrm{I}\Sigma_1}{\vdash} \ell(\ulcorner s \urcorner) = 0^{((s \dot{-} \mathbf{1})_1)},$$

pero esto es lo mismo que $\underset{\mathrm{I}\Sigma_1}{\vdash} \ell(\ulcorner s \urcorner) = 0^{(\ell(s))}$.

La definición 2.5 se traduce a los teoremas siguientes de $\mathrm{I}\Sigma_1$:

$$\underset{\mathrm{I}\Sigma_1}{\vdash} R(s, 0) = (s \dot{-} 0^{(\mathbf{1})})_2, \qquad \underset{\mathrm{I}\Sigma_1}{\vdash} R(s, i + 0^{(\mathbf{1})}) = R(s, i)_1.$$

En particular, usando los resultados precedentes,

$$\underset{\mathrm{I}\Sigma_1}{\vdash} R(\ulcorner s \urcorner, 0^{(0)}) = 0^{((s \dot{-} \mathbf{1})_2)}, \qquad \underset{\mathrm{I}\Sigma_1}{\vdash} R(\ulcorner s \urcorner, 0^{(i+\mathbf{1})}) = R(\ulcorner s \urcorner, 0^{(i)})_1.$$

A partir de aquí, una inducción sobre i prueba que

$$\bigwedge i \underset{\mathrm{I}\Sigma_1}{\vdash} R(\ulcorner s \urcorner, 0^{(i)}) = 0^{(R(s,i))}.$$

En efecto, como $R(s,\mathbf{0}) = (s\dot{-}\mathbf{1})_2$, el primero de los dos teoremas precedentes equivale al caso $i = \mathbf{0}$ y, si es válido para i, el segundo teorema equivale a

$$\underset{\mathrm{I}\Sigma_1}{\vdash} R(\ulcorner s\urcorner, 0^{(i+\mathbf{1})}) = 0^{(R(s,i)_1)},$$

y basta tener en cuenta que $R(s, i+\mathbf{1}) = R(s,i)_1$ para tener la conclusión. Finalmente, la definición de $p_i^\infty(s)$ dada en 2.5 nos da que

$$\underset{\mathrm{I}\Sigma_1}{\vdash} p_i^\infty(s) = (1\dot{-}i)R(s,(s\dot{-}1)_1) + (1\dot{-}(1\dot{-}i))R(s,(s\dot{-}1)_1\dot{-}i)_2.$$

Al particularizar a $0^{(i)}, \ulcorner s\urcorner$ y aplicar los resultados precedentes obtenemos que

$$\underset{\mathrm{I}\Sigma_1}{\vdash} p^\infty_{0^{(i)}}(\ulcorner s\urcorner) = 0^{(p_i^\infty(s))},$$

que es lo mismo que $\underset{\mathrm{I}\Sigma_1}{\vdash} \ulcorner s\urcorner_{0^{(i)}} = 0^{(s_i)}$. ∎

Como consecuencia:

Teorema 6.8 $\bigwedge st \underset{\mathrm{I}\Sigma_1}{\vdash} \ulcorner s\urcorner^\frown \ulcorner t\urcorner = \ulcorner s^\frown t\urcorner$.

Demostración: Pongamos que $\ell(s) = m, \ell(t) = n$. Entonces

$$\underset{\mathrm{I}\Sigma_1}{\vdash} (\ell(\ulcorner s\urcorner) = 0^{(m)} \wedge \ell(\ulcorner t\urcorner) = 0^{(n)} \wedge \ell(\ulcorner s^\frown t\urcorner) = 0^{(m+n)}).$$

Por otro lado, $\underset{\mathrm{I}\Sigma_1}{\vdash} \ell(x^\frown y) = \ell(x) + \ell(y)$, luego $\underset{\mathrm{I}\Sigma_1}{\vdash} \ell(\ulcorner s\urcorner^\frown \ulcorner t\urcorner) = 0^{(m)} + 0^{(n)}$. Por lo tanto, $\underset{\mathrm{I}\Sigma_1}{\vdash} \ell(\ulcorner s\urcorner^\frown \ulcorner t\urcorner) = \ell(\ulcorner s^\frown t\urcorner)$.

Si $i < m$, tenemos que $(s^\frown t)_i = s_i$, luego $\underset{\mathrm{I}\Sigma_1}{\vdash} \ulcorner s^\frown t\urcorner_{0^{(i)}} = 0^{(s_i)}$.

Por otro lado, $\underset{\mathrm{I}\Sigma_1}{\vdash} 0^{(i)} < \ell(\ulcorner s\urcorner)$, luego, por la definición de $^\frown$, tenemos

$$\underset{\mathrm{I}\Sigma_1}{\vdash} (\ulcorner s\urcorner^\frown \ulcorner t\urcorner)_{0^{(i)}} = \ulcorner s\urcorner_{0^{(i)}},$$

luego $\underset{\mathrm{I}\Sigma_1}{\vdash} (\ulcorner s\urcorner^\frown \ulcorner t\urcorner)_{0^{(i)}} = \ulcorner s^\frown t\urcorner_{0^{(i)}}$. Si $m \le i < m+n$ llegamos análogamente a la misma conclusión, luego esto es válido para todo $i < m+n$.

Por último, el teorema 6.4 nos da (suponiendo que $m+n \neq \mathbf{0}$)

$$\underset{\mathrm{I}\Sigma_1}{\vdash} i < 0^{(m+n)} \to i = 0^{(\mathbf{0})} \vee \cdots \vee i = 0^{(m+n\dot{-}\mathbf{1})},$$

con lo que las igualdades previas implican que

$$\underset{\mathrm{I}\Sigma_1}{\vdash} i < 0^{(m+n)}\ (\ulcorner s\urcorner^\frown \ulcorner t\urcorner)_i = \ulcorner s^\frown t\urcorner_i$$

(y si $m+n = \mathbf{0}$ se cumple también trivialmente). Ahora basta usar que

$$\underset{\mathrm{I}\Sigma_1}{\vdash} \ell(s) = \ell(t) \wedge \bigwedge i < \ell(s) s_i = t_i \to s = t$$

para llegar a la conclusión. ∎

Si aplicamos el teorema 6.7 a una sucesión s que sea, concretamente, una cadena de signos de $\mathcal{L}_a$, entonces s_i es su signo i-ésimo, y el numeral $0^{(s_i)}$ es su formalización. Por ejemplo si

$$s \equiv x = 0 = \langle =, x, 0\rangle = \langle \mathbf{4}, \mathbf{16}, \mathbf{0}\rangle = \mathbf{269\,619\,034},$$

tenemos que $s_{\mathbf{0}} \equiv = \equiv \mathbf{4}$, con lo que $0^{(s\mathbf{0})} = 4 = \ulcorner = \urcorner$, por lo que en este caso es más natural escribir $\ulcorner s_i\urcorner$ en lugar de $0^{(s_i)}$ (ya que no pensamos en $0^{(s_i)}$ como en un número natural, sino como en un signo de $\ulcorner\mathcal{L}_a\urcorner$).

En estos términos, (un caso particular de) el teorema 6.7 se expresa así:

Si $\zeta \in \mathrm{Cad}(\mathcal{L}_a)$, *entonces* $\underset{\mathrm{I}\Sigma_1}{\vdash} \ell(\ulcorner\zeta\urcorner) = 0^{(\ell(\zeta))}$ *y, para cada* $i < \ell(\zeta)$, *se cumple*

$$\underset{\mathrm{I}\Sigma_1}{\vdash} (\ulcorner\zeta\urcorner)_{0^{(i)}} = \ulcorner\zeta_i\urcorner.$$

Esto, junto con el teorema 6.8, permite probar sistemáticamente una larga serie de resultados que relacionan $\mathcal{L}_a$ con $\ulcorner\mathcal{L}_a\urcorner$. Por ejemplo:

Si $\theta \in \mathrm{STerm}(\mathcal{L}_a)$, *entonces* $\underset{\mathrm{I}\Sigma_1}{\vdash} \ulcorner\theta\urcorner \in \mathrm{STerm}(\ulcorner\mathcal{L}_a\urcorner)$,

Si $\theta \in \mathrm{SForm}(\mathcal{L}_a)$, *entonces* $\underset{\mathrm{I}\Sigma_1}{\vdash} \ulcorner\theta\urcorner \in \mathrm{SForm}(\ulcorner\mathcal{L}_a\urcorner)$.

Esto se prueba por inducción sobre ζ, sin más que usar que en la teoría $\mathrm{I}\Sigma_1$ (no en la metateoría) se puede probar el teorema 3.5 que caracteriza las semiexpresiones, así como las consecuencias de la definición de $\mathcal{L}_a$.

Por ejemplo, en el caso en que $\theta \equiv \langle x\rangle$ es (una cadena cuyo único signo es) una variable libre, tenemos que $x \equiv \mathbf{2}^{i+\mathbf{4}}$, para cierto i, luego[4] $\underset{\mathrm{I}\Sigma_1}{\vdash} \ulcorner x\urcorner = 2^{0^{(i)}+4}$, y ahora usamos que esto implica que $\ulcorner x\urcorner \in \mathrm{VarLib}(\ulcorner\mathcal{L}_a\urcorner)$ por la formalización de la definición de $\mathcal{L}_a$. El teorema 6.7 nos da que $\underset{\mathrm{I}\Sigma_1}{\vdash} \ulcorner\theta\urcorner = \langle\ulcorner x\urcorner\rangle$ y usando la formalización del teorema 3.5 concluimos que $\underset{\mathrm{I}\Sigma_1}{\vdash} \ulcorner\theta\urcorner \in \mathrm{STerm}(\ulcorner\mathcal{L}_a\urcorner)$.

Otro ejemplo: si $\theta \equiv \bigwedge u\alpha$, donde $u \in \mathrm{VLig}(\mathcal{L}_a)$ y $\alpha \in \mathrm{Form}(\mathcal{L}_a)$, como antes concluimos que $\underset{\mathrm{I}\Sigma_1}{\vdash} \ulcorner u\urcorner \in \mathrm{VLig}(\ulcorner\mathcal{L}_a\urcorner)$ y, por hipótesis de inducción,

$$\underset{\mathrm{I}\Sigma_1}{\vdash} \ulcorner\alpha\urcorner \in \mathrm{SForm}(\ulcorner\mathcal{L}_a\urcorner).$$

Además, como $\theta \equiv \langle\bigwedge, u\rangle^\frown \alpha$, el teorema 6.8 nos da que

$$\underset{\mathrm{I}\Sigma_1}{\vdash} \ulcorner\theta\urcorner = \langle\ulcorner\textstyle\bigwedge\urcorner, \ulcorner u\urcorner\rangle^\frown\ulcorner\alpha\urcorner,$$

y la formalización del teorema 3.5 implica que

$$\underset{\mathrm{I}\Sigma_1}{\vdash} \ulcorner\theta\urcorner \in \mathrm{SForm}(\ulcorner\mathcal{L}_a\urcorner).$$

Los demás casos se tratan análogamente. ∎

[4] Aquí usamos que $\bigwedge mn \underset{\mathrm{I}\Sigma_1}{\vdash} (0^{(m)})^{0^{(n)}} = 0^{(m^n)}$, lo cual se prueba trivialmente por inducción sobre n, usando 6.2.

Del mismo modo se prueba, por ejemplo, que[5]

$$\mathrm{VarLib}(\theta) = \{x_1, \ldots, x_n\} \to \underset{\mathrm{I}\Sigma_1}{\vdash} \mathrm{VarLib}(\ulcorner\theta\urcorner) = \{\ulcorner x_1\urcorner, \ldots, \ulcorner x_n\urcorner\},$$

donde $\mathrm{VarLib}(\theta)$ representa el conjunto de variables que aparecen libres en la expresión θ.

Con esto tenemos todo lo necesario para demostrar —o, mejor dicho, para entender cabalmente el enunciado— de los teoremas principales de la sección siguiente, pero antes vamos a discutir una sutileza adicional.

Ejemplo Para profundizar en la relación entre los tres niveles de lenguaje que estamos considerando, tomemos por ejemplo la fórmula de $\mathcal{L}_a^0$.

$$x = 0 \equiv = x_0 0$$

Su formalización en $\mathcal{L}_a$ es $\alpha \equiv x = 0 \equiv \langle \mathbf{4}, \mathbf{16}, \mathbf{0}\rangle = \mathbf{269\,619\,034}$, y a su vez podemos considerar su formalización en $\ulcorner\mathcal{L}_a\urcorner$, que es[6]

$$\ulcorner\alpha\urcorner = \ulcorner x = 0\urcorner = \ulcorner\mathbf{269\,619\,034}\urcorner = 269\,619\,034 = \langle 4, 16, 0\rangle\,.$$

Observemos que α es un numeral de $\mathcal{L}_a^0$, luego en particular no tiene variables libres, lo cual no contradice que en el $\mathrm{I}\Sigma_1$ metamatemático podemos demostrar que $\alpha \in \mathrm{Form}(\mathcal{L}_a)$ es una fórmula con $x \equiv \mathbf{16}$ como única variable libre.

A su vez, $\ulcorner\alpha\urcorner$ es otro término de $\mathcal{L}_a^0$, de modo que en el $\mathrm{I}\Sigma_1$ metamatemático se demuestra que $\ulcorner\alpha\urcorner \in \mathrm{Term}(\mathcal{L}_a)$, y es un designador (un numeral, sin variables libres), así como que

$$\underset{\mathrm{I}\Sigma_1}{\vdash} \ulcorner\alpha\urcorner \in \mathrm{Form}(\ulcorner\mathcal{L}_a\urcorner)$$

y, más concretamente, en el $\mathrm{I}\Sigma_1$ formalizado se demuestra que es una fórmula con $\ulcorner x\urcorner \equiv \ulcorner\mathbf{16}\urcorner \equiv 16$ como única variable libre.

La sutileza a la que hacíamos referencia aparece al considerar, por una parte, el término $\ulcorner z\urcorner$ de $\mathcal{L}_a^0$, con z como única variable libre, determinado por los teoremas (metamatemáticos):

$$\ulcorner\mathbf{0}\urcorner = 0, \qquad \ulcorner z + \mathbf{1}\urcorner = (\ulcorner z\urcorner)'.$$

[5]Para que la inducción sea respecto de una fórmula Σ_1 hay que hacer algunos trucos. Si llamamos $u_1, \ldots, u_m$ a las variables que están ligadas en θ, podemos probar por inducción sobre θ (con las x_i, u_j fijas) que, llamando Var al conjunto de todas las variables que aparecen en una cadena de signos,

$$\mathrm{Var}(\theta) \subset \{x_1, \ldots, x_n, u_1, \ldots, u_m\} \to \underset{\mathrm{I}\Sigma_1}{\vdash} \mathrm{Var}(\ulcorner\theta\urcorner) \subset \{\ulcorner x_1\urcorner, \ldots, \ulcorner x_n\urcorner, \ulcorner u_1\urcorner, \ldots, \ulcorner u_m\urcorner\},$$

y luego probar (también por inducción sobre θ) que

$$x_i \in \mathrm{VarLib}(\theta) \to \underset{\mathrm{I}\Sigma_1}{\vdash} \ulcorner x_i\urcorner \in \mathrm{VarLib}(\ulcorner\theta\urcorner), \quad u_i \notin \mathrm{VarLib}(\theta) \to \underset{\mathrm{I}\Sigma_1}{\vdash} \ulcorner u_i\urcorner \notin \mathrm{VarLib}(\theta).$$

[6]Si quisiéramos expresarla como un numeral de $\mathcal{L}_a^0$, sería el que codifica la sucesión que consta de 269 619 034 unos seguidos de un cero, un número más que astronómico.

Según hemos visto, este término cumple (entre otros resultados similares):

$$x \in \mathrm{Var}(\mathcal{L}_a) \to \underset{\mathrm{I}\Sigma_1}{\vdash} \ulcorner x \urcorner \in \mathrm{Var}(\ulcorner \mathcal{L}_a \urcorner), \quad \alpha \in \mathrm{Form}(\mathcal{L}_a) \to \underset{\mathrm{I}\Sigma_1}{\vdash} \ulcorner \alpha \urcorner \in \mathrm{Form}(\ulcorner \mathcal{L}_a \urcorner).$$

Por ejemplo, si $x \equiv \mathbf{16} \in \mathrm{Var}(\mathcal{L}_a)$, entonces $\ulcorner x \urcorner \equiv 16$ cumple

$$\underset{\mathrm{I}\Sigma_1}{\vdash} 16 \in \mathrm{Var}(\ulcorner \mathcal{L}_a \urcorner).$$

Por otra parte, podemos considerar el designador $0^{(x)}$ de $\mathcal{L}_a^0$ del que se demuestra (a nivel metamatemático) que $0^{(x)} \in \mathrm{Term}(\mathcal{L}_a)$ es un término con x como única variable libre, y además

$$\underset{\mathrm{I}\Sigma_1}{\vdash} 0^{(0)} = 0 \wedge \bigwedge x\, 0^{(x+1)} = (0^{(x)})'.$$

Así, si $x \in \mathrm{Var}(\mathcal{L}_a)$, no debemos confundir $\ulcorner x \urcorner$ con $0^{(x)}$. En el caso concreto en que $x = \mathbf{16}$, tenemos que $\ulcorner x \urcorner = 16$ es una variable de $\ulcorner \mathcal{L}_a \urcorner$, mientras que $0^{(x)}$ es el término de $\mathcal{L}_a$ con x como variable libre tal que $0^{(0)}, 0^{(1)}, 0^{(2)}, \ldots$ son los numerales de $\ulcorner \mathcal{L}_a \urcorner$.

Notemos que podríamos escribir $\ulcorner x \urcorner \equiv 0^{(x)}$, pero sería un $0^{(x)}$ distinto del que estamos considerando aquí (de ahí la conveniencia de usar dos notaciones distintas), $\ulcorner x \urcorner \equiv 0^{(x)}$ es el término de $\mathcal{L}_a^0$ que enumera los numerales de $\mathcal{L}_a$, mientras que el $0^{(x)}$ que estamos considerando aquí es su formalización, el término $\ulcorner \ulcorner x \urcorner \urcorner \equiv \ulcorner 0^{(x)} \urcorner$ que enumera los numerales de $\ulcorner \mathcal{L}_a \urcorner$. ∎

6.3 Satisfacción de fórmulas de $\mathcal{L}_a$

Hasta ahora hemos visto que en ARP (luego en $\mathrm{I}\Sigma_1$) podemos formalizar toda la parte sintáctica de la lógica de primer orden, pero no hemos dicho nada sobre los resultados en los que intervienen modelos. De hecho, en la sección 3.1 vimos que no es posible formalizar en ARP el concepto de "número natural denotado por un término de $\mathcal{L}_{\mathrm{arp}}$ respecto de una valoración", lo cual nos impedía a su vez formalizar el concepto de satisfacción de una fórmula respecto de una valoración y, en general, todos los resultados relacionados con la interpretación de las fórmulas. Ahora vamos a ver que en $\mathrm{I}\Sigma_1$ podemos formalizar una parte sustancial de la semántica de $\mathcal{L}_a$. Entre otras cosas, esto nos permitirá demostrar que las teorías $\mathrm{I}\Sigma_n$ son finitamente axiomatizables, es decir, que (aparte de los infinitos axiomas lógicos) basta una cantidad finita de sus axiomas (propios) para demostrar todos los demás.

De momento, podemos marcarnos como objetivo definir una fórmula $\mathbb{N} \vDash \alpha$ en $\mathcal{L}_a$ de tal modo que, para toda sentencia α de $\mathcal{L}_a$, se cumpla

$$\mathbb{N} \vDash \alpha \leftrightarrow \mathbb{N} \vDash (\mathbb{N} \vDash \ulcorner \alpha \urcorner).$$

(Notemos que aquí las dos primeras "$\mathbb{N} \vDash$" representan el concepto metamatemático de "verdad" en el modelo natural de $\mathcal{L}_a$, definido en 3.3.) En realidad

esto exactamente es imposible, pero obtendremos algo aproximado, y lo más interesante que el lector puede extraer de aquí es hacerse una idea de los inconvenientes que impiden en la práctica formalizar los conceptos semánticos con la misma facilidad con que pueden formalizarse los conceptos sintácticos.

Observemos que para cada $\alpha \in \mathrm{Form}(\ulcorner\mathcal{L}_a\urcorner)$ se cumple que α es una $\ulcorner$fórmula$\urcorner$, pero a la vez es un término (en principio, una mera variable de $\ulcorner\mathcal{L}_a\urcorner$). En particular, sería absurdo escribir algo así como

$$\bigwedge\alpha \in \mathrm{Form}(\ulcorner\mathcal{L}_a\urcorner)\ (\alpha \to \alpha).$$

No es que esto sea falso, sino que es incoherente. Si hay que entender $\alpha \to \alpha$ como $\alpha \ulcorner\to\urcorner \alpha$, entonces $\alpha \ulcorner\to\urcorner \alpha \equiv \langle\ulcorner\to\urcorner\rangle \frown \alpha \frown \alpha$ es un término de $\ulcorner\mathcal{L}_a\urcorner$, no una fórmula, por lo que es asintáctico escribirlo tras $\bigwedge\alpha \in \mathrm{Form}(\mathcal{L}_a)$. Si hay que entender $\to$ como el implicador metamatemático, entonces es asintáctico ponerlo entre dos variables.

Esto es el equivalente formal del hecho de que las fórmulas de un lenguaje formal no significan nada (no son realmente afirmaciones) hasta que no se fija un modelo de su lenguaje, y eso es lo que vamos a hacer ahora, al menos en un caso particular.

Definición 6.9 Diremos que v es una *valoración* de una semiexpresión θ (y lo representaremos por $\mathrm{Val}(v,\theta)$) si v es una función definida sobre un conjunto (finito) de variables de $\ulcorner\mathcal{L}_a\urcorner$ que incluye al menos a todas las que aparecen libres en θ. Es claro que $\mathrm{Val}(v,\theta)$ es una fórmula de $\mathcal{L}_a$ de tipo Δ_1.

Definimos también:

$$v_x^a = (v \setminus \{(x, v(x))\}) \cup \{(x,a)\},$$

de modo que si $\mathrm{Val}(v,\theta)$ y x es una variable, entonces v_x^a es otra valoración de θ que coincide con v salvo a lo sumo en x, donde toma el valor a (aunque en principio no es preciso que v esté definida en x).

Teorema 6.10 *Existe un término* $\mathrm{Dn}(t,v)$ *de* $\mathcal{L}_a$ *de tipo* Δ_1 *en* $\mathrm{I}\Sigma_1$ *tal que en dicha teoría se demuestran las propiedades siguientes:*

1. *Si* x *es una variable y* $\mathrm{Val}(v,x)$*, entonces* $\mathrm{Dn}(x,v) = v(x)$*,*
2. $\mathrm{Dn}(0,v) = 0$*,*
3. *Si* t *es un semitérmino y* $\mathrm{Val}(v,t)$*, entonces* $\mathrm{Dn}(St,v) = \mathrm{Dn}(t,v) + 1$*,*
4. *Si* t_1 *y* t_2 *son semitérminos y* $\mathrm{Val}(v,t_1)$*,* $\mathrm{Val}(v,t_2)$*, entonces*

$$\mathrm{Dn}(t_1 + t_2, v) = \mathrm{Dn}(t_1,v) + \mathrm{Dn}(t_2,v) \wedge \mathrm{Dn}(t_1 \cdot t_2, v) = \mathrm{Dn}(t_1,v) \cdot \mathrm{Dn}(t_2,v).$$

Notemos que en la igualdad $\mathrm{Dn}(t_1 + t_2, v) = \mathrm{Dn}(t_1,v) + \mathrm{Dn}(t_2,v)$ el primer $+$ es el funtor $\ulcorner+\urcorner$ de $\ulcorner\mathcal{L}_a\urcorner$, mientras que el segundo es el funtor $+$ de $\mathcal{L}_{\mathrm{arp}}$. Lo mismo vale para el producto.

DEMOSTRACIÓN: Por comodidad vamos a definir un funtor diádico Dn en ARP que cumpla lo requerido, de modo que la fórmula $n = \mathrm{Dn}(t, v)$ de $\mathcal{L}_{\mathrm{arp}}$ se traducirá a una fórmula Δ_1 de $\mathcal{L}_a$ que a su vez nos permite definir (como una descripción) el término $\mathrm{Dn}(t, v)$ requerido.

Para ello definimos por recurrencia un funtor $\mathrm{Dn}^*(V, t, v)$ que cumple las propiedades anteriores con la exigencia adicional (en todos los casos) de que $\mathrm{Var}(t) \subset V \subset \mathcal{D}v$, donde $\mathrm{Var}(t)$ es el conjunto de todas las variables que aparecen en t. Así basta definir $\mathrm{Dn}^*(V, t, v)$ supuesto que está definido para todo $t_0 \in \mathrm{Sub}(t)$, según el teorema 2.23.

Luego se prueba por inducción que si v y v' son valoraciones con V en su dominio y que coinciden sobre el conjunto de variables que aparecen libres en t y éste está contenido en V, entonces $\mathrm{Dn}^*(V, t, v) = \mathrm{Dn}^*(V, t, v')$, lo que a su vez permite definir $\mathrm{Dn}(t, v) = \mathrm{Dn}^*(\mathrm{Var}(t), t, v^*)$, donde v^* es la valoración que resulta de extender v a cualquier variable de $\mathrm{Var}(t) \setminus \mathcal{D}v$ en la que no estuviera definida asignándole el valor 0 (por ejemplo). Se prueba entonces que $\mathrm{Dn}(t, v)$ cumple todo lo requerido. ∎

Nota Observemos que, modificando de forma obvia la prueba del teorema anterior, podemos definir $\mathrm{Dn}(t, v)$ para el caso en que t es un semitérmino del lenguaje $\ulcorner\mathcal{L}_a^*\urcorner$ definido en 6.5, añandiendo para ello al enunciado la condición $\mathrm{Dn}(\mathrm{pre}\, t, v) = \mathrm{Dn}(t, v) \dot{-} 1$. ∎

Completando el argumento de la demostración anterior, es fácil ver que si v y v' son dos valoraciones que coinciden sobre las variables libres de un mismo semitérmino t, entonces $\mathrm{Dn}(t, v) = \mathrm{Dn}(t, v')$. Si t es un designador escribiremos $\mathrm{Dn}(t) \equiv \mathrm{Dn}(t, \varnothing)$.

En la prueba del teorema siguiente necesitaremos el hecho siguiente: si v y w son dos valoraciones tales que $\mathrm{Val}(v, t)$ y $\mathrm{Val}(w, t)$ y además, cuando ambas están definidas en una misma variable x se cumple $v(x) \leq w(x)$, entonces se cumple $\mathrm{Dn}(t, v) \leq \mathrm{Dn}(t, w)$.

Esto se debe esencialmente a que tanto la suma como el producto son operaciones monótonas (si sustituimos unos sumandos/factores por otros mayores, el valor de la suma/producto aumenta, y esto sigue siendo cierto si consideramos semitérminos de $\ulcorner\mathcal{L}_a^*\urcorner$, con el funtor pre).

Teorema 6.11 *Existe una fórmula $\mathbb{N} \vDash_0 \alpha[v]$ de tipo Δ_1 en $\mathrm{I}\Sigma_1$ tal que en dicha teoría se demuestra que, si α es una semifórmula de tipo Δ_0 y $\mathrm{Val}(v, \alpha)$,*

1. *Si $\alpha \equiv t_1 = t_2$, entonces $\mathbb{N} \vDash_0 (t_1 = t_2)[v] \leftrightarrow \mathrm{Dn}(t_1, v) = \mathrm{Dn}(t_2, v)$,*
2. *Si $\alpha \equiv t_1 \leq t_2$, entonces $\mathbb{N} \vDash_0 (t_1 \leq t_2)[v] \leftrightarrow \mathrm{Dn}(t_1, v) \leq \mathrm{Dn}(t_2, v)$,*
3. *Si $\alpha \equiv \neg\beta$, entonces $\mathbb{N} \vDash_0 \neg\beta[v] \leftrightarrow \neg\mathbb{N} \vDash_0 \beta[v]$,*
4. *Si $\alpha \equiv \beta \vee \gamma$, entonces $\mathbb{N} \vDash_0 (\beta \vee \gamma)[v] \leftrightarrow (\mathbb{N} \vDash_0 \beta[v] \vee \mathbb{N} \vDash_0 \gamma[v])$,*

5. *Si* $\alpha \equiv \bigwedge u \le t\,\beta$, *entonces*

$$\mathbb{N} \vDash_0 (\bigwedge u \le t\,\alpha)[v] \leftrightarrow \bigwedge a \le \mathrm{Dn}(t, v)\ \mathbb{N} \vDash_0 \alpha[v^a_u],$$

6. *Si* $\alpha \equiv \bigvee u \le t\,\beta$, *entonces*

$$\mathbb{N} \vDash_0 (\bigvee u \le t\,\alpha)[v] \leftrightarrow \bigvee a \le \mathrm{Dn}(t, v)\ \mathbb{N} \vDash_0 \alpha[v^a_u].$$

Demostración: Fijemos una semifórmula α_0 de tipo Δ_0 y una valoración v_0 definida únicamente sobre las variables que aparecen libres en α_0, digamos $x_1, \ldots, x_m$, con lo que en particular cumple $\mathrm{Val}(v_0, \alpha_0)$. Sean $u_1, \ldots, u_n$ las variables que aparecen ligadas en α_0 en el orden en que lo hacen, de izquierda a derecha. Cada una aparecerá en la forma $\bigwedge u_i \le t_i$ o $\bigvee u_i \le t_i$ para cierto semitérmino t_i cuyas variables libres serán a lo sumo $x_1, \ldots, x_n, u_1, \ldots, u_{i-1}$. Definimos recurrentemente:

$$c(v_0, \alpha_0, i) = \mathrm{Dn}(t_i, (v_0)^{c(v_0,\alpha_0,1),\ldots,c(v_0,\alpha_0,i-1)}_{u_1,\ldots,u_{i-1}}).$$

Llamamos $V(\alpha_0, v_0)$ al conjunto de todas las valoraciones v definidas sobre las variables x_i, u_i tales que $v(x_i) = v_0(x_i)$, $v(u_i) \le c(v_0, \alpha_0, i)$.

La observación previa a este teorema muestra que si $v \in V(\alpha_0, v_0)$, entonces $\mathrm{Dn}(t_i, v) \le c(v_0, \alpha_0, i)$.

En efecto, como t_i sólo tiene libres las variables $x_1, \ldots x_m, u_1, \ldots u_{i-1}$, tenemos que $\mathrm{Dn}(t_i, v) = \mathrm{Dn}(t_i, v')$, donde v' es la restricción de v a estas variables y, para ellas tenemos que

$$v'(x_i) = v_0(x_i) = (v_0)^{c(v_0,\alpha_0,1),\ldots,c(v_0,\alpha_0,i-1)}_{u_1,\ldots,u_{i-1}}(x_i),$$

$$v'(u_i) = v(u_i) \le c(v_0, \alpha_0, i) = (v_0)^{c(v_0,\alpha_0,1),\ldots,c(v_0,\alpha_0,i-1)}_{u_1,\ldots,u_{i-1}}(u_i),$$

luego

$$\mathrm{Dn}(t_i, v) = \mathrm{Dn}(t_i, v') \le \mathrm{Dn}(t_i, (v_0)^{c(v_0,\alpha_0,1),\ldots,c(v_0,\alpha_0,i-1)}_{u_1,\ldots,u_{i-1}}) = c(v_0, \alpha_0, i).$$

Vamos a definir un funtor $\mathrm{Sat}_0(\alpha_0, v_0, \alpha)$ tal que

$$\mathrm{Sat}_0(\alpha_0, v_0, \alpha) : V(\alpha_0, v_0) \longrightarrow \{0, 1\}$$

y de modo que si α es una subsemifórmula de α_0 y $v \in V(\alpha_0, v_0)$, se cumple:

1. Si $\alpha \equiv t_1 = t_2$, entonces $\mathrm{Sat}_0(\alpha_0, v_0, \alpha)(v) = 1 \leftrightarrow \mathrm{Dn}(t_1, v) = \mathrm{Dn}(t_2, v)$,
2. Si $\alpha \equiv t_1 \le t_2$, entonces $\mathrm{Sat}_0(\alpha_0, v_0, \alpha)(v) = 1 \leftrightarrow \mathrm{Dn}(t_1, v) \le \mathrm{Dn}(t_2, v)$,
3. Si $\alpha \equiv \neg\beta$, entonces $\mathrm{Sat}_0(\alpha_0, v_0, \alpha)(v) = 1 \leftrightarrow \mathrm{Sat}_0(\alpha_0, v_0, \beta)(v) = 0$,
4. Si $\alpha \equiv \beta \vee \gamma$, entonces

$$\mathrm{Sat}_0(\alpha_0, v_0, \alpha)(v) = 1 \leftrightarrow \mathrm{Sat}_0(\alpha_0, v_0, \beta)(v) = 1 \vee \mathrm{Sat}_0(\alpha_0, v_0, \gamma)(v) = 1,$$

5. Si $\alpha \equiv \bigwedge u \leq t\, \beta$, entonces

$$\mathrm{Sat}_0(\alpha_0, v_0, \alpha)(v) = 1 \leftrightarrow \bigwedge a \leq \mathrm{Dn}(t, v)\, \mathrm{Sat}_0(\alpha_0, v_0, \beta)(v_u^a) = 1,$$

6. Si $\alpha \equiv \bigvee u \leq t\, \beta$, entonces

$$\mathrm{Sat}_0(\alpha_0, v_0, \alpha)(v) = 1 \leftrightarrow \bigvee a \leq \mathrm{Dn}(t, v)\, \mathrm{Sat}_0(\alpha_0, v_0, \beta)(v_u^a) = 1.$$

Basta aplicar el teorema 2.23 para definir $\mathrm{Sat}_0(\alpha_0, v_0, \alpha)$ supuesto que ya está definido $\mathrm{Sat}_0(\alpha_0, v_0, \beta)$ para todo $\beta \in \mathrm{Sub}(\alpha)$. Notemos que en los dos últimos apartados es fundamental que si α es una subsemifórmula de α_0, necesariamente $u \equiv u_i$ y $t \equiv t_i$ para algún i, con lo que si $a \leq \mathrm{Dn}(t_i, v) \leq c(v_0, \alpha_0, i)$, podemos asegurar que $v_{u_i}^a \in V(\alpha_0, v_0)$.

Una simple inducción prueba que si $v \in V(\alpha_0, v_0)$ y $v' \in V(\alpha_0, v_0')$ son dos valoraciones que coinciden sobre las variables que están libres en una subsemifórmula α de α_0, entonces

$$\mathrm{Sat}_0(\alpha_0, v_0, \alpha)(v) = \mathrm{Sat}_0(\alpha_0, v_0', \alpha)(v').$$

La razón de fondo es que en el cálculo de Sat_0 no se usa v_0 en ningún momento, sino que ésta sólo es necesaria para restringir la recursión de modo que sólo suponga definidos un número finito de valores del funtor.

Consideramos entonces un funtor $v_0(\alpha_0, v)$ que a cada valoración v definida sobre las variables que aparecen libres en la subsemifórmula α de α_0 le asigna una valoración definida sobre las variables libres de α_0 que coincida con v cuando ésta esté definida, así como otro funtor $w(\alpha_0, v)$ que extienda $v_0(\alpha_0, v)$ a todas las variables de α_0 y coincida con v sobre las que ésta esté definida. Así podemos definir

$$\mathrm{Sat}_0^*(\alpha_0, \alpha, v) = \mathrm{Sat}_0(\alpha_0, v_0(\alpha_0, v), \alpha)(w(\alpha_0, v)),$$

de modo que se siguen cumpliendo las propiedades de Sat_0.

Igualmente, en el cálculo de Sat_0 o Sat_0^* no se usa en ningún momento α_0, por lo que si α es una subsemifórmula de dos semifórmulas α_0, α_0', también tenemos la igualdad $\mathrm{Sat}_0^*(\alpha_0, \alpha, v) = \mathrm{Sat}_0^*(\alpha_0', \alpha, v)$. Por lo tanto, podemos definir

$$\mathbb{N} \vDash_0 \alpha[v] \equiv \mathrm{Sat}_0^*(\alpha, \alpha, v) = 1,$$

y se cumplen todos los requisitos del enunciado. Por ejemplo,

$$\mathbb{N} \vDash_0 \neg\beta[v] \leftrightarrow \mathrm{Sat}_0^*(\neg\beta, \neg\beta, v) = 1 \leftrightarrow \mathrm{Sat}_0^*(\neg\beta, \beta, v) = 0 \leftrightarrow$$
$$\mathrm{Sat}_0^*(\beta, \beta, v) = 0 \leftrightarrow \neg\mathbb{N} \vDash_0 \beta[v]. \qquad \blacksquare$$

De las propiedades del teorema anterior se siguen las consecuencias obvias:

$$\mathbb{N} \vDash_0 (\alpha \wedge \beta)[v] \leftrightarrow \mathbb{N} \vDash_0 \alpha[v] \wedge \mathbb{N} \vDash_0 \beta[v],$$
$$\mathbb{N} \vDash_0 (\alpha \to \beta)[v] \leftrightarrow \neg\mathbb{N} \vDash_0 \alpha[v] \vee \mathbb{N} \vDash_0 \beta[v],$$

etc.

Destacamos el teorema siguiente, cuya prueba está implícita en la del teorema anterior:

Teorema 6.12 *Si v y v' son valoraciones que coinciden sobre las variables libres de una fórmula α de tipo Δ_0, entonces* $\mathbb{N} \vDash_0 \alpha[v] \leftrightarrow \mathbb{N} \vDash_0 \alpha[v']$.

Todos estos resultados valen igualmente, con las mismas pruebas, para fórmulas de $\ulcorner\mathcal{L}_a^*\urcorner$.

No es posible definir en $\mathrm{I}\Sigma_1$ (ni siquiera en AP) una fórmula $\mathbb{N} \vDash \alpha[v]$ que formalice el concepto de satisfacción para fórmulas arbitrarias, pero podemos ir un poco más lejos:

Definición 6.13 Para cada número natural n, definimos recurrentemente las fórmulas siguientes de $\mathcal{L}_a$:

$$\mathbb{N} \vDash_{\Sigma_0} \alpha[v] \equiv \mathbb{N} \vDash_0 \alpha[v], \quad \mathbb{N} \vDash_{\Pi_0} \alpha[v] \equiv \neg\mathbb{N} \vDash_0 \neg\alpha[v],$$

$$\mathbb{N} \vDash_{\Sigma_{n+1}} \alpha[v] \equiv \alpha \in \Sigma_n \wedge \mathbb{N} \vDash_{\Sigma_n} \alpha[v] \vee \alpha \in \Pi_n \wedge \mathbb{N} \vDash \alpha[v] \vee$$

$$\bigvee \beta u(\beta \in \Pi_n \wedge u \in \mathrm{VarLig}(\ulcorner\mathcal{L}_a\urcorner)$$

$$\wedge \alpha = \bigvee u\, \beta \wedge \mathrm{Val}(v, \alpha) \wedge \bigvee m\ \mathbb{N} \vDash_{\Pi_n} \beta[v_u^m]).$$

$$\mathbb{N} \vDash_{\Pi_{n+1}} \alpha[v] \equiv \alpha \in \Sigma_n \wedge \mathbb{N} \vDash_{\Sigma_n} \alpha[v] \vee \alpha \in \Pi_n \wedge \mathbb{N} \vDash \alpha[v] \vee$$

$$\bigvee \beta u(\beta \in \Sigma_n \wedge u \in \mathrm{VarLig}(\ulcorner\mathcal{L}_a\urcorner)$$

$$\wedge \alpha = \bigwedge u\, \beta \wedge \mathrm{Val}(v, \alpha) \wedge \bigwedge m\ \mathbb{N} \vDash_{\Sigma_n} \beta[v_u^m]),$$

donde $u \in \mathrm{VarLig}\ulcorner\mathcal{L}_a\urcorner$ es la fórmula que expresa que u es una variable ligada del lenguaje $\ulcorner\mathcal{L}_a\urcorner$. Omitiremos v cuando sea $v = \varnothing$.

Es muy importante destacar que con esto no hemos definido dos fórmulas $\mathbb{N} \vDash_{\Sigma_n} \alpha[v]$, $\mathbb{N} \vDash_{\Pi_n} \alpha[v]$ de $\mathcal{L}_a$ con tres variables libres (n, α, v), sino infinitas fórmulas (dos para cada n) con dos variables libres (α, v).

Por inducción se prueba que la fórmula $\mathbb{N} \vDash_{\Sigma_n} \alpha[v]$ es Σ_n, mientras que $\mathbb{N} \vDash_{\Pi_n} \alpha[v]$ es Π_n (para $n \geq 1$). Notemos únicamente que el cuantificador existencial de $\mathbb{N} \vDash_{\Pi_{n+1}} \alpha[v]$ puede cambiarse por

$$\bigwedge z(z = \mathcal{R}\alpha \cup (\mathcal{R}\alpha)^{<\ell(\alpha)} \to \bigvee \beta u \in z \cdots).$$

Ahora vamos a probar que estas fórmulas expresan lo que cabe esperar que expresen. Primero necesitamos un resultado para términos.

Teorema 6.14 *Sea $t(u_1, \ldots, u_r)$ un semitérmino de $\mathcal{L}_a$ sin descriptores con las variables libres indicadas (todas ligadas). Entonces, en $\mathrm{I}\Sigma_1$ se demuestra:*

$$\bigwedge u_1 \cdots u_r\ t(u_1, \ldots, u_r) = \mathrm{Dn}(\ulcorner t\urcorner, \{(\ulcorner u_1\urcorner, u_1), \ldots, (\ulcorner u_r\urcorner, u_r)\}).$$

DEMOSTRACIÓN: Por abreviar, llamemos $v \equiv \{(\ulcorner u_1 \urcorner, u_1), \ldots, (\ulcorner u_r \urcorner, u_r)\}$. Notemos que $\mathrm{Var}(\ulcorner t \urcorner) = \{\ulcorner u_1 \urcorner, \ldots, \ulcorner u_r \urcorner\}$, luego se cumple $\mathrm{Val}(v, \ulcorner t \urcorner)$. Vamos a construir la demostración pedida para cada uno de los subsemitérminos t_0 de t.

Si $t_0 \equiv u_i$, entonces $\mathrm{Dn}(\ulcorner u_i \urcorner, v) = v(\ulcorner u_i \urcorner) = u_i = t_0$, luego se cumple el teorema.

Si $t_0 \equiv 0$, entonces $\mathrm{Dn}(0, v) = 0 = t_0$.

Si $t_0 \equiv St_1$ y el teorema es cierto para t_1, entonces se puede probar que $\mathrm{Dn}(\ulcorner t_1 \urcorner, v) = t_1$, y la prueba se puede extender hasta $\mathrm{Dn}(\ulcorner St_1 \urcorner, v) = t_1 + 1 = St_1$.

Si $t_0 \equiv t_1 + t_2$ y podemos demostrar que $\mathrm{Dn}(\ulcorner t_1 \urcorner, v) = t_1$ y $\mathrm{Dn}(\ulcorner t_2 \urcorner, v) = t_2$, y la prueba se puede extender hasta

$$\mathrm{Dn}(\ulcorner t_1 + t_2 \urcorner, v) = \mathrm{Dn}(\ulcorner t_1 \urcorner, v) + \mathrm{Dn}(\ulcorner t_2 \urcorner, v) = t_1 + t_2 = t_0.$$

El caso $t_0 \equiv t_1 \cdot t_2$ es análogo. ∎

En particular, para designadores tenemos que $t = \mathrm{Dn}(\ulcorner t \urcorner)$.

Teorema 6.15 *Sea $\phi(u_1, \ldots, u_r)$ una semiformula Σ_n de $\mathcal{L}_a$ con las variables libres indicadas (todas ligadas). Entonces, en $\mathrm{I}\Sigma_1$ se demuestra:*

$$\bigwedge u_1 \cdots u_r \; (\phi(u_1, \ldots, u_r) \leftrightarrow \; \mathbb{N} \vDash_{\Sigma_n} \ulcorner \phi \urcorner[\{(\ulcorner u_1 \urcorner, u_1), \ldots, (\ulcorner u_r \urcorner, u_r)\}]).$$

Lo mismo vale cambiando Σ_n por Π_n.

DEMOSTRACIÓN: Llamemos $v \equiv \{(\ulcorner u_1 \urcorner, u_1), \ldots, (\ulcorner u_r \urcorner, u_r)\}$. Consideramos en primer lugar el caso $n = 0$, es decir, que ϕ es Δ_0. Razonamos por inducción[7] sobre la longitud de ϕ. Si $\phi \equiv t_1 = t_2$, entonces, usando 6.14 vemos que

$$t_1 = t_2 \leftrightarrow \mathrm{Dn}(\ulcorner t_1 \urcorner, v) = \mathrm{Dn}(\ulcorner t_2 \urcorner, v) \leftrightarrow \mathbb{N} \vDash_0 (\ulcorner t_1 = t_2 \urcorner, v).$$

El caso $t_1 \le t_2$ es análogo. Si $\phi \equiv \neg\psi$, entonces, por hipótesis de inducción podemos probar

$$\psi \leftrightarrow \mathbb{N} \vDash_0 \ulcorner \psi \urcorner[v],$$

luego

$$\phi \leftrightarrow \neg\mathbb{N} \vDash_0 \ulcorner \psi \urcorner[v] \leftrightarrow \mathbb{N} \vDash_0 \ulcorner \neg\psi \urcorner[v] \leftrightarrow \mathbb{N} \vDash_0 \ulcorner \phi \urcorner[v].$$

El caso $\phi \equiv \psi \vee \chi$ es similar. Si $\phi \equiv \bigwedge u_0 \le t\, \psi$, entonces por hipótesis de inducción,

$$\bigwedge u_0 \cdots u_n \psi(u_0, \ldots, u_n) \leftrightarrow \mathbb{N} \vDash_0 \ulcorner \psi \urcorner[v^{u_0}_{\ulcorner u_0 \urcorner}],$$

luego

$$\bigwedge u \le t\, \psi \leftrightarrow \bigwedge u \le \mathrm{Dn}(t, v)\, \mathbb{N} \vDash_0 \ulcorner \psi \urcorner[v^{u_0}_{\ulcorner u_0 \urcorner}] \leftrightarrow \mathbb{N} \vDash_0 \ulcorner \phi \urcorner[v].$$

El caso en que $\phi \equiv \bigvee u_0 \le t\, \psi$ es análogo.

[7] Más precisamente, si llamamos $\Psi(\phi)$ a la fórmula del enunciado, se trata de probar que $\bigwedge \phi(\phi \in \Delta_0 \to \vdash_{\ulcorner \mathrm{I}\Sigma_1 \urcorner} \Psi(\phi))$. La inducción sobre ϕ es formalizable en ARP, pues podemos definir un funtor D que a cada semifórmula ϕ le asigne la demostración requerida $D(\phi)$ suponiéndolo definido sobre las semifórmulas de $\mathrm{Sub}(\phi)$.

Esto termina la prueba en el caso $n = 0$. En el caso general, si ϕ es una fórmula de tipo Σ_n o Π_n, será de la forma $\pi\,\alpha(u_1,\dots,u_r,v_1,\dots,v_n)$, donde π es una sucesión de cuantificadores alternados que ligan las variables $v_1,\dots,v_n$. Razonamos por inducción[8] sobre i que la semifórmula

$$\phi_i \equiv \pi_i\alpha(u_1,\dots,u_r,v_1,\dots,v_n),$$

en la que sólo se han ligado las variables $v_{n-i+1},\dots,v_n$, cumple el enunciado, es decir, que si, por ejemplo, $\phi_{i+1} \equiv \bigvee v_{n-i}\phi_i$, donde ϕ_i es Π_i (la alternativa en la que el cuantificador es universal y ϕ_i es Σ_i se trata análogamente), entonces en $\mathrm{I}\Sigma_1$ se demuestra

$$\phi_i(u_1,\dots,u_r,v_1,\dots,v_{n-i}) \leftrightarrow \mathbb{N} \vDash_{\Pi_i} {}^{\ulcorner}\phi_i^{\urcorner}[v^{v_1,\ \dots,\ v_{n-i}}_{\ulcorner v_1\urcorner,\dots,\ulcorner v_{n-i}\urcorner}].$$

Para $i = 0$ es el caso $n = 0$ ya probado. Si lo suponemos cierto para i,

$$\phi_{i+1} \leftrightarrow \bigvee u_{n-i}\,\mathbb{N} \vDash_{\Pi_i} {}^{\ulcorner}\phi_i^{\urcorner}[v^{v_1,\ \dots,\ v_{n-(i+1)},\ v_{n-i}}_{\ulcorner v_1\urcorner,\dots,\ulcorner v_{n-(i+1)}\urcorner,\ulcorner v_{n-i}\urcorner}] \leftrightarrow$$

$$\mathbb{N} \vDash_{\Sigma_{n+1}} {}^{\ulcorner}\phi^{\urcorner}[v^{v_1,\ \dots,\ v_{n-(i+1)}}_{\ulcorner v_1\urcorner,\dots,\ulcorner v_{n-(i+1)}\urcorner}].$$

Para $i = n$ tenemos la equivalencia del enunciado. ∎

Teniendo en cuenta que $\mathrm{Dn}(0^{(y)}) = y$, una inducción rutinaria muestra, más en general, que si $t(x_1,\dots,x_r)$ es un semitérmino de ${}^{\ulcorner}\mathcal{L}_a{}^{\urcorner}$, entonces

$$\mathrm{Dn}(t(x_1,\dots,x_r),\{(x_1,y_1),\dots(x_r,y_r)\}) = \mathrm{Dn}(t(0^{(y_1)},\dots,0^{(y_n)})),$$

así como que si $\phi(x_1,\dots,x_r)$ es una semifórmula de ${}^{\ulcorner}\mathcal{L}_a{}^{\urcorner}$,

$$\mathbb{N} \vDash_{\Sigma_n} \phi(x_1,\dots,x_r),\{(x_1,y_1),\dots(x_r,y_r)\}) \leftrightarrow \mathbb{N} \vDash_{\Sigma_n} \phi(0^{(y_1)},\dots,0^{(y_r)}),$$

(y lo mismo vale para $\mathbb{N} \vDash_{\Pi_n}$), lo que nos permite enunciar de una forma más cómoda los dos teoremas precedentes:

Teorema 6.16 *Si $t(x_1,\dots,x_r)$ es un término de $\mathcal{L}_a$ y $\phi(x_1,\dots,x_r)$ es una fórmula de tipo Σ_n (o Π_n) cuyas variables libres están entre las indicadas, entonces en $\mathrm{I}\Sigma_1$ se demuestra:*

$$\bigwedge x_1\cdots x_r\ t(x_1,\dots,x_r) = \mathrm{Dn}({}^{\ulcorner}t^{\urcorner}(0^{(x_1)},\dots,0^{(x_r)}),$$

$$\bigwedge x_1\cdots x_r\ (\phi(x_1,\dots,x_r) \leftrightarrow\ \mathbb{N} \vDash_{\Sigma_n} {}^{\ulcorner}\phi^{\urcorner}(0^{(x_1)},\dots,0^{(x_r)}))$$

(o con $\mathbb{N} \vDash_{\Pi_n}$ si ϕ es de tipo Π_n.) En particular, si ϕ es una sentencia de $\mathcal{L}_a$, se cumple que

$$\underset{\mathrm{I}\Sigma_1}{\vdash} (\phi \leftrightarrow \mathbb{N} \vDash_{\Sigma_n} {}^{\ulcorner}\phi^{\urcorner})$$

(o con $\mathbb{N} \vDash_{\Pi_n}$ si ϕ es de tipo Π_n).

[8] Nuevamente, probamos por inducción sobre i que la fórmula $\Psi(n,i,\alpha)$ es demostrable en ${}^{\ulcorner}\mathrm{I}\Sigma_1{}^{\urcorner}$. Técnicamente, podemos considerar que definimos por recurrencia sobre i un funtor $F(n,i,\alpha)$ que determina una demostración en ${}^{\ulcorner}\mathrm{I}\Sigma_1{}^{\urcorner}$ de la fórmula $\Psi(n,i,\alpha)$.

6.4 La Σ_1-completitud de Q

Ahora podemos demostrar una propiedad fundamental de la aritmética de Robinson, y que es la razón principal por la que ésta tiene interés, y es que bastan los débiles axiomas de Q para demostrar cualquier afirmación de tipo Σ_1 que sea verdadera:

Teorema 6.17 (de Σ_1-completitud de Q) *Si $\alpha(x_1,\ldots,x_n)$ es una fórmula de tipo Σ_1 en $\ulcorner\mathcal{L}_a\urcorner$ cuyas variables libres están entre las indicadas y $\mathrm{Val}(v,\alpha)$, entonces*

$$\mathbb{N}\models_{\Sigma_1}\alpha[v]\to\underset{\ulcorner Q\urcorner}{\vdash}\alpha(0^{(v(x_1))},\ldots,0^{(v(x_n))}).$$

En particular, si α es una sentencia de tipo Σ_1, se cumple $\mathbb{N}\models_{\Sigma_1}\alpha\to\underset{\ulcorner Q\urcorner}{\vdash}\alpha$.

Demostración: En primer lugar probamos que si $t(x_1,\ldots,x_n)$ es un semitérmino de $\ulcorner\mathcal{L}_a\urcorner$ sin descriptores, entonces

$$\mathrm{Val}(v,t)\to\underset{\ulcorner Q\urcorner}{\vdash}0^{(\mathrm{Dn}(t,v))}=t(0^{(v(x_1))},\ldots,0^{(v(x_n))}).$$

La fórmula es Σ_1 y podemos razonar por inducción sobre t.

Si $t\equiv x$ es una variable (libre o ligada), suponemos que v está definida en x y tenemos que probar que

$$\underset{\ulcorner Q\urcorner}{\vdash}0^{(v(x))}=0^{(v(x))},$$

lo cual es obvio. El caso $t\equiv 0$ es similar.

Si $t\equiv t_0'$, por hipótesis de inducción tenemos que

$$\underset{\ulcorner Q\urcorner}{\vdash}0^{(\mathrm{Dn}(t_0,v))}=t_0(0^{(v(x_1))},\ldots,0^{(v(x_n))}),$$

y la demostración se puede prolongar para obtener

$$\underset{\ulcorner Q\urcorner}{\vdash}(0^{(\mathrm{Dn}(t_0,v))})'=t(0^{(v(x_1))},\ldots,0^{(v(x_n))}),$$

y, por definición de $0^{(n)}$,

$$\underset{\ulcorner Q\urcorner}{\vdash}0^{(\mathrm{Dn}(t_0,v)+1)}=t(0^{(v(x_1))},\ldots,0^{(v(x_n))}),$$

y, finalmente, por definición de Dn,

$$\underset{\ulcorner Q\urcorner}{\vdash}0^{(\mathrm{Dn}(t,v))}=t(0^{(v(x_1))},\ldots,0^{(v(x_n))}).$$

Los casos $t\equiv t_1+t_2$ y $t\equiv t_1\cdot t_2$ son similares, y con ellos se concluye la prueba de esta primera parte.

Ahora, como en la prueba del teorema 6.11, fijamos una semifórmula α_0 de tipo Δ_0 y una valoración v_0 tal que $\mathrm{Val}(v_0, \alpha_0)$ definida únicamente sobre las variables que están libres en α_0 y consideramos el mismo conjunto de valoraciones $V(\alpha_0, v_0)$ que allí, las cuales están definidas sobre todas las variables $x_1, \ldots, x_n$ de α_0, libres o ligadas.[9] En esta ocasión definimos un funtor $D(\alpha_0, v_0, \alpha)$ tal que si α es una subsemifórmula de α_0, entonces $D(\alpha_0, v_0, \alpha)$ es una función que a cada valoración $v \in V(\alpha_0, v_0)$ le asigna una demostración $D(\alpha_0, v_0, \alpha)(v) = d$, de modo que

$$((\mathbb{N} \vDash_0 \alpha[v] \to \overset{d}{\underset{\ulcorner Q \urcorner}{\vdash}} \alpha(0^{(v(x_1))}, \ldots, 0^{(v(x_n))})) \wedge$$

$$(\neg\mathbb{N} \vDash_0 \alpha[v] \to \overset{d}{\underset{\ulcorner Q \urcorner}{\vdash}} \neg\alpha(0^{(v(x_1))}, \ldots, 0^{(v(x_n))}))).$$

En principio tendríamos que definir el funtor D (por recurrencia, definiéndolo para α supuesto que está definido para semifórmulas en $\mathrm{Sub}(\alpha)$) y luego demostrar por inducción sobre α que cumple este hecho, pero, como la inducción motiva la definición, las presentamos simultáneamente.

Si $\alpha \equiv t_1 = t_2$, llamamos $a_i = \mathrm{Dn}(t_i, v)$, de modo que, por el resultado precedente,

$$\underset{\ulcorner Q \urcorner}{\vdash} 0^{(a_i)} = t_i(0^{(v(x_1))}, \ldots, 0^{(v(x_n))}).$$

Más aún, la prueba de este resultado es constructiva, por lo que tenemos un funtor $D(t)$ que nos da la demostración en Q correspondiente al término t. Por otro lado, la condición $\mathbb{N} \vDash_0 \alpha[v]$ se particulariza en este caso a que $a_1 = a_2$. Es claro entonces, en virtud del teorema 6.2, que podemos definir $D(\alpha_0, v_0, \alpha)$ que cumpla lo requerido, según si $a_1 = a_2$ o $a_1 \neq a_2$.

Si $\alpha \equiv \neg\beta$ basta tomar $D(\alpha_0, v_0, \alpha)(v) = D(\alpha_0, v_0, \beta)(v)$ si $\neg\mathbb{N} \vDash_0 \beta[v]$ y, en caso contrario, $D(\alpha_0, v_0, \alpha)(v)$ se obtiene prolongando la demostración $D(\alpha_0, v_0, \beta)(v)$ para pasar de β a $\neg\neg\beta \equiv \neg\alpha$.

Si $\alpha \equiv \beta \vee \gamma$, si se cumple $\mathbb{N} \vDash_0 \beta[v]$, o bien $\mathbb{N} \vDash_0 \gamma[v]$ o bien $\mathbb{N} \vDash_0 \gamma[v]$. Llamando $d = D(\alpha_0, v_0, \beta)(v)$ o bien $d = D(\alpha_0, v_0, \gamma)(v)$, según el caso, la hipótesis de inducción nos da que

$$\overset{d}{\underset{\ulcorner Q \urcorner}{\vdash}} \beta(0^{(v(x_1))}, \ldots, 0^{(v(x_n))}) \vee \overset{d}{\underset{\ulcorner Q \urcorner}{\vdash}} \gamma(0^{(v(x_1))}, \ldots, 0^{(v(x_n))}),$$

y la demostración se prolonga claramente hasta una demostración en Q de la sentencia $\alpha(0^{(v(x_1))}, \ldots, 0^{(v(x_n))})$.

Si, por el contrario, $\neg\mathbb{N} \vDash_0 \beta[v]$ y $\neg\mathbb{N} \vDash_0 \gamma[v]$, entonces la hipótesis de inducción nos da demostraciones $d_1 = D(\alpha_0, v_0, \beta)(v)$ y $d_2 = D(\alpha_0, v_0, \gamma)(v)$ tales que

$$\overset{d_1}{\underset{\ulcorner Q \urcorner}{\vdash}} \neg\beta(0^{(v(x_1))}, \ldots, 0^{(v(x_n))}) \wedge \overset{d_2}{\underset{\ulcorner Q \urcorner}{\vdash}} \neg\gamma(0^{(v(x_1))}, \ldots, 0^{(v(x_n))}),$$

con las cuales podemos formar a su vez una demostración en Q de la negación $\neg\alpha(0^{(v(x_1))}, \ldots, 0^{(v(x_n))})$.

[9] Por conveniencia, aquí llamamos $x_1, \ldots, x_n$ a todas las variables de α_0, que no es la misma notación que usábamos en la demostración de 6.11.

Si $\alpha \equiv \bigwedge x \leq t\, \beta$ y $\mathbb{N} \vDash_0 \alpha(v)$, esto significa que

$$\bigwedge i \leq \mathrm{Dn}(t, v)\, \mathbb{N} \vDash_0 \beta[v^i_x],$$

y en este punto es crucial que, como razonamos en la prueba de 6.11, en estas condiciones las valoraciones v^i_x están todas en $V(\alpha_0, v_0)$, por lo que tenemos definidas las demostraciones $d_i = D(\alpha_0, v_0, \beta)(v^i_x)$ tales que

$$\underset{\ulcorner Q \urcorner}{\overset{d_i}{\vdash}} \beta(0^{(i)}, 0^{(v(x_1))}, \ldots, 0^{(v(x_n))}).$$

Todas ellas se combinan para probar

$$\underset{\ulcorner Q \urcorner}{\vdash} y = 0^{(0)} \vee \cdots \vee y = 0^{(\mathrm{Dn}(t,v))} \to \beta(y, 0^{(v(x_1))}, \ldots, 0^{(v(x_n))}).$$

Usando 6.4, concluimos que

$$\underset{\ulcorner Q \urcorner}{\vdash} \bigwedge u \leq 0^{(\mathrm{Dn}(t,v))}\, \beta(u, 0^{(v(x_1))}, \ldots, 0^{(v(x_n))}),$$

lo cual, por la primera parte de la demostración, equivale a

$$\underset{\ulcorner Q \urcorner}{\vdash} \bigwedge u \leq t(0^{(v(x_1))}, \ldots, 0^{(v(x_n))})\, \beta(u, 0^{(v(x_1))}, \ldots, 0^{(v(x_n))}).$$

Si, por el contrario, $\neg\mathbb{N} \vDash_0 \alpha(v)$, entonces existe un $i \leq \mathrm{Dn}(t, v)$ de manera que $\neg\mathbb{N} \vDash_0 \beta[v^i_u]$, y la hipótesis de inducción nos da una demostración

$$\underset{\ulcorner Q \urcorner}{\overset{d}{\vdash}} \neg\beta(0^{(i)}, 0^{(v(x_1))}, \ldots, 0^{(v(x_n))}),$$

que puede prolongarse sucesivamente a una demostración en Q de

$$\bigvee u \leq 0^{(\mathrm{Dn}(t,v))} \neg\beta(u, 0^{(v(x_1))}, \ldots, 0^{(v(x_n))})$$

y a su vez de $\neg\alpha(0^{(v(x_1))}, \ldots, 0^{(v(x_n))})$.

El caso en que $\alpha \equiv \bigvee u \leq x\, \beta$ se trata análogamente.

Si ahora tenemos en cuenta que $\mathbb{N} \vDash_0 \alpha[v]$ sólo depende de los valores que v toma sobre las variables libres en α, podemos definir

$$D^*(\alpha_0, \alpha, v) = D(\alpha_0, v_0(\alpha_0, v), \alpha)(w(\alpha_0, v)),$$

donde v es cualquier valoración definida sobre las variables que están libres en α y los funtores v_0 y w determinan las valoraciones requeridas por el funtor D, exactamente igual que en la demostración de 6.11. Finalmente, si α y v cumplen $\mathrm{Val}(v, \alpha)$, la demostración $D^*(\alpha, \alpha, v)$ atestigua que el teorema es correcto. ∎

Combinando este teorema con 6.15 obtenemos la versión siguiente:

Teorema 6.18 (de Σ_1-completitud de Q) *Si $\alpha(x_1, \ldots, x_n)$ es una fórmula de $\mathcal{L}_a$ de tipo Σ_1 cuyas únicas variables libres estén entre las variables $x_1, \ldots, x_n$ y $\ulcorner\alpha\urcorner(\ulcorner x_1\urcorner, \ldots, \ulcorner x_n\urcorner)$ es su formalización en $\ulcorner\mathcal{L}_a\urcorner$, entonces*

$$\underset{\mathrm{I}\Sigma_1}{\vdash} (\alpha(x_1, \ldots, x_n) \to \underset{\ulcorner Q \urcorner}{\vdash} \ulcorner\alpha\urcorner(0^{(x_1)}, \ldots, 0^{(x_n)})).$$

En particular, si α es una sentencia Σ_1 de $\mathcal{L}_a$, se cumple que

$$\underset{\mathrm{I}\Sigma_1}{\vdash} (\alpha \to \underset{\ulcorner Q \urcorner}{\vdash} \ulcorner\alpha\urcorner).$$

Nota En principio, el teorema anterior se prueba formalmente en la metateoría $\mathrm{I}\Sigma_1$ y se aplica a fórmulas $\alpha \in \mathrm{Form}(\mathcal{L}_a)$, pero, como todo razonamiento formal en $\mathrm{I}\Sigma_1$, puede verse también como un esquema teoremático aplicable a cada fórmula metamatemática α de $\mathcal{L}_a^0$ y a su formalización $\ulcorner\alpha\urcorner$ en $\mathcal{L}_a$, de modo que la implicación

$$\alpha(x_1, \ldots, x_n) \to \underset{Q}{\vdash} \ulcorner\alpha\urcorner(0^{(x_1)}, \ldots, 0^{(x_n)})).$$

se demuestra en la metateoría $\mathrm{I}\Sigma_1$.

En esencia, esto significa que si unos números naturales cumplen una fórmula α de tipo Σ_1, esto se puede demostrar en Q, con la precisión técnica necesaria de que lo que se demuestra en Q es la formalización de α en el lenguaje formalizado $\mathcal{L}_a$.

Así el teorema 6.2 es un caso particular del teorema de Σ_1-completitud. Por ejemplo, el apartado 1. puede expresarse como que

$$r = m + n \to \underset{Q}{\vdash} 0^{(r)} = 0^{(m)} + 0^{(n)}.$$

Más aún, también los teoremas 6.7 y 6.8 pueden verse ahora como casos particulares del teorema de Σ_1-completitud, lo que permite mejorarlos, ya que las sentencias pueden demostrarse en Q en lugar de en $\mathrm{I}\Sigma_1$. Igualmente podemos concluir que en $\mathrm{I}\Sigma_1$ se demuestran los resultados del estilo de

$$t \in \mathrm{Term}(\mathcal{L}_a) \to \underset{\mathrm{Q}}{\vdash} \ulcorner t\urcorner \in \mathrm{Term}(\ulcorner\mathcal{L}_a\urcorner),$$

$$\alpha \in \mathrm{Form}(\mathcal{L}_a) \to \underset{\mathrm{Q}}{\vdash} \ulcorner\alpha\urcorner \in \mathrm{Form}(\ulcorner\mathcal{L}_a\urcorner),$$

que conectan la metateoría con su formalización (notemos que, por definición, $\ulcorner t\urcorner \equiv 0^{(t)}$, $\ulcorner\alpha\urcorner \equiv 0^{(\alpha)}$ son los numerales de $\ulcorner\mathcal{L}_a\urcorner$ correspondientes a t, α vistos como números naturales). Nos interesa especialmente el caso de la fórmula

$$\underset{\mathrm{I}\Sigma_1}{\vdash} \alpha \equiv \bigvee d \overset{d}{\underset{\mathrm{I}\Sigma_1}{\vdash}} \alpha.$$

La versión metamatemática del teorema de Σ_1-completitud nos da que

$$\underset{\mathrm{I}\Sigma_1}{\vdash} \alpha \to \underset{Q}{\vdash} \underset{\ulcorner\mathrm{I}\Sigma_1\urcorner}{\vdash} \ulcorner\alpha\urcorner.$$

(para todo α, entendiendo que la definición de $\underset{\mathrm{I}\Sigma_1}{\vdash} \alpha$ exige que $\alpha \in \mathrm{Form}(\mathcal{L}_a)$). Nuevamente observamos que $\ulcorner\alpha\urcorner \equiv 0^{(\alpha)}$, por lo que la implicación anterior se ajusta ciertamente al enunciado del teorema de Σ_1-completitud. ∎

Ya casi podemos probar que $\mathrm{I}\Sigma_n$ es finitamente axiomatizable, pero para ello necesitamos examinar con más detalle la demostración del teorema 6.15 (lo cual, a su vez, nos lleva también a la de 6.14) para comprobar que, aunque se trata de esquemas teoremáticos, que afirman que una familia infinita de fórmulas son demostrables en $\mathrm{I}\Sigma_1$, todas ellas se pueden probar a partir de un mismo conjunto finito de axiomas de $\mathrm{I}\Sigma_1$.

En lo que sigue es fundamental que cuando decimos "teorema" queremos decir propiamente "teorema" y no "esquema teoremático". Todo teorema es demostrable a partir de un número finito de axiomas, mientras que, en principio, un esquema teoremático podría requerir infinitos axiomas, un conjunto distinto para demostrar cada uno de sus casos particulares.

Vamos a encontrar un conjunto finito Γ de axiomas de $\mathrm{I}\Sigma_1$ tal que a partir de Γ se puedan probar todos los casos particulares del teorema 6.14. En principio incluimos en Γ todos los axiomas de Q más los necesarios para demostrar los axiomas de la teoría básica de conjuntos B, que son también un número finito (teorema 5.20), e iremos añadiendo los que necesitemos a lo largo de la demostración.

Nos ocupamos primero del semitérmino $v \equiv \{(\ulcorner u_1 \urcorner, u_1), \ldots, (\ulcorner u_n \urcorner, u_n)\}$. Observamos que, si llamamos D_n a la conjunción de todas las fórmulas $y_i \neq y_j$, para $i \neq j$, el esquema teoremático

$$\bigwedge v_1 \cdots v_n u_1 \cdots u_n (D_n \to \{(v_1, u_1), \ldots, (v_n, y_n)\} \text{ es una función}$$
$$\wedge\, \mathcal{D}(\{(v_1, u_1), \ldots, (v_n, u_n)\}) = \{v_1, \ldots, v_n\}),$$

se demuestra a partir de los axiomas de B, luego también a partir de Γ. Eliminando los generalizadores, obtenemos el esquema teoremático:

$$\bigwedge u_1 \cdots u_n (D^* \to \{(\ulcorner u_1 \urcorner, u_1) \ldots, (\ulcorner u_n \urcorner, u_n)\} \text{ es una función} \wedge$$
$$\mathcal{D}(\{(\ulcorner u_1 \urcorner, u_1) \ldots, (\ulcorner u_n \urcorner, u_n)\}) = \{\ulcorner u_1 \urcorner, \ldots, \ulcorner u_n \urcorner\}),$$

donde D^* es la conjunción de las sentencias $\ulcorner u_i \urcorner \neq \ulcorner u_j \urcorner$ (para $i \neq j$), que es, pues, demostrable a partir de Γ.

Más aún, si $u_1, \ldots, u_n$ son variables de $\mathcal{L}_a$ distintas dos a dos, por 6.2 tenemos que $\ulcorner u_i \urcorner \neq \ulcorner u_j \urcorner$ es demostrable en Q, luego la conjunción de todas estas sentencias es demostrable a partir de Γ, luego concluimos que si $u_1, \ldots, u_n$ son variables cualesquiera distintas dos a dos,

$$\underset{\Gamma}{\vdash} \bigwedge u_1 \cdots u_n (\{(\ulcorner u_1 \urcorner, u_1) \ldots, (\ulcorner u_n \urcorner, u_n)\} \text{ es una función} \wedge$$
$$\mathcal{D}(\{(\ulcorner u_1 \urcorner, u_1) \ldots, (\ulcorner u_n \urcorner, u_n)\}) = \{\ulcorner u_1 \urcorner, \ldots, \ulcorner u_n \urcorner\}).$$

Ahora consideramos la fórmula[10] Σ_1

$$x \in \mathrm{STerm}(\mathcal{L}_a) \wedge \mathrm{VarLib}(x) \subset \{x_1, \ldots, x_n\}.$$

El teorema de Σ_1-completitud 6.18 nos da que

$$x \in \mathrm{STerm}(\mathcal{L}_a) \wedge \mathrm{VarLib}(x) \subset \{x_1, \ldots, x_n\} \to$$
$$\underset{Q}{\vdash}\ 0^{(x)} \in \mathrm{STerm}^{\ulcorner}\mathcal{L}_a^{\urcorner} \wedge \mathrm{VarLib}(0^{(x)}) \subset \{0^{(x_1)}, \ldots, 0^{(x_n)}\}.$$

[10]Hay que entender que es una fórmula del lenguaje $\mathcal{L}_a^0$ correspondiente a la teoría $\mathrm{I}\Sigma_1$ que usamos como metateoría, en la que se demuestra 6.14. Se trata de una fórmula Δ_1, es decir, equivalente en $\mathrm{I}\Sigma_1$ tanto a una fórmula Σ_1 como a otra Π_1. Entendemos que la fórmula que consideramos aquí es precisamente la versión Σ_1.

Si aplicamos este hecho al término t de las hipótesis de 6.14 y a las variables $u_1, \ldots, u_n \in \mathrm{Var}(\mathcal{L}_a)$, obtenemos[11]

$$\underset{Q}{\vdash} \ulcorner t \urcorner \in \mathrm{STerm}\ulcorner \mathcal{L}_a \urcorner \wedge \mathrm{VarLib}(\ulcorner t \urcorner) \subset \{\ulcorner u_1 \urcorner, \ldots, \ulcorner u_n \urcorner\}.$$

En particular, esta fórmula se demuestra a partir de los axiomas de Γ y, uniendo esto a lo anterior, tenemos que

$$\underset{\Gamma}{\vdash} \bigwedge u_1 \cdots u_n \mathrm{Val}(v, \ulcorner t \urcorner).$$

Esto implica que v y $\ulcorner t \urcorner$ están en las hipótesis del teorema 6.10. Extendiendo Γ, podemos suponer que este teorema también se prueba a partir de Γ, así como el esquema teoremático

$$\ulcorner s \urcorner^\frown \ulcorner t \urcorner = \ulcorner s^\frown t \urcorner,$$

que se demuestra en Q. La demostración del teorema 6.14 no requiere nada más. Por ejemplo, si podemos probar a partir de Γ que $\mathrm{Dn}(\ulcorner t_1 \urcorner, v) = t_1$ y $\mathrm{Dn}(\ulcorner t_2 \urcorner, v) = t_2$, entonces también podemos probar que

$$\ulcorner t_1 + t_2 \urcorner = \ulcorner + \urcorner^\frown \ulcorner t_1 \urcorner^\frown \ulcorner t_2 \urcorner,$$

lo que a su vez nos permite aplicar 6.10 para concluir que

$$\mathrm{Dn}(\ulcorner t_1 + t_2 \urcorner, v) = \mathrm{Dn}(\ulcorner t_1 \urcorner, v) + \mathrm{Dn}(\ulcorner t_2 \urcorner, v) = t_1 + t_2.$$

Lo mismo vale para todos los otros casos de la inducción.

Ahora pasamos a analizar la demostración de 6.15 y empezamos por el caso $n = 0$. Exactamente igual que en el análisis precedente podemos concluir que, añadiendo axiomas a Γ, podemos contar con que a partir de Γ se demuestra

$$\mathrm{Val}(\{(\ulcorner u_1 \urcorner, u_1), \ldots (\ulcorner u_r \urcorner, u_r), (\ulcorner v_1 \urcorner, v_1), \ldots, (\ulcorner v_n \urcorner, v_n)\}, \ulcorner \alpha \urcorner)$$

en todos los casos en que la demostración lo requiere, así como el teorema 6.11. Con esto, lo único que falta para justificar que todas las demostraciones del caso $n = 0$ pueden hacerse a partir de Γ es un pequeño detalle técnico: Γ permite probar que si u_0 es una variable distinta de $u_1, \ldots, u_n$, entonces

$$v^{u_0}_{\ulcorner u_0 \urcorner} = \{(\ulcorner u_0 \urcorner, u_0), (\ulcorner u_1 \urcorner, u_1), \ldots, (\ulcorner u_n \urcorner, u_n)\},$$

pero esto puede probarse claramente en la teoría de conjuntos B, luego también a partir de Γ.

Para probar el caso general sólo necesitamos añadir a Γ los axiomas necesarios para demostrar las definiciones de las fórmulas Σ_n y Π_n, así como las definiciones 6.13 de $\mathbb{N} \vDash_{\Sigma_i} \alpha[v]$ y $\mathbb{N} \vDash_{\Pi_i} \alpha[v]$ para $i = 1, \ldots, n$ (son definiciones recursivas, luego en realidad son teoremas que aseguran que las fórmulas cumplen lo requerido).

[11] Notemos que, al sustituir x por t en $0^{(x)}$ obtenemos el numeral $0^{(t)}$ en $\ulcorner \mathcal{L}_a \urcorner$ correspondiente al número t, que es lo que en este contexto llamamos $\ulcorner t \urcorner$. Lo mismo vale con $0^{(u_i)} = \ulcorner u_i \urcorner$.

Si a partir de Γ se demuestra la fórmula del caso i-ésimo de la inducción, para demostrar el caso $i+1$-ésimo sólo se usa la hipótesis de inducción, la definición 6.13 y algunas manipulaciones de valoraciones, todas ellas demostrables en B. ∎

Finalmente podemos probar:

Teorema 6.19 *Para todo $n > 0$, la teoría $\mathrm{I}\Sigma_n$ es finitamente axiomatizable, es decir, todos sus teoremas pueden probarse a partir de un conjunto finito de axiomas (propios).*

Demostración: Consideremos la fórmula

$$\mathbb{N} \vDash_{\Sigma_n} \phi[v_x^0] \wedge \bigwedge m(\mathbb{N} \vDash_{\Sigma_n} \phi[v_x^m] \to \mathbb{N} \vDash_{\Sigma_n} \phi[v_x^{m+1}]) \to \bigwedge m\ \mathbb{N} \vDash_{\Sigma_n} \phi[v_x^m], \quad (6.1)$$

donde ϕ y x son variables libres. Se trata de un caso particular del principio de Σ_n-inducción, luego es un axioma de $\mathrm{I}\Sigma_n$.

Fijemos un conjunto finito Γ de axiomas de $\mathrm{I}\Sigma_n$ que contenga al axioma anterior y permita demostrar el teorema 6.11 (que es un único teorema) y el teorema 6.15 (que es un esquema teoremático, pero ya hemos justificado que es demostrable a partir de un número finito de axiomas), así como los resultados auxiliares a los que hemos hecho referencia al justificar que 6.15 requiere sólo un conjunto finito de axiomas.

Basta probar que a partir de Γ se pueden demostrar todos los casos particulares del principio de inducción con fórmulas Σ_n. En efecto, si $\phi(x, x_1, \ldots, x_n)$ es una fórmula Σ_n y abreviamos $v \equiv \{(\ulcorner x_1 \urcorner, x_1), \ldots, (\ulcorner x_n \urcorner, x_n)\}$, a partir de Γ podemos probar el teorema 6.15, que nos da la equivalencia

$$\phi \leftrightarrow \mathbb{N} \vDash_{\Sigma_n} \ulcorner \phi \urcorner [v_{\ulcorner x \urcorner}^x]. \quad (6.2)$$

Notemos que ambas partes tienen libres las variables $x_1, \ldots, x_n, x$. Por el axioma (6.1) tenemos (generalizando respecto de las variables ϕ y x y luego sustituyendo por los designadores $\ulcorner \phi \urcorner$ y $\ulcorner x \urcorner$):

$$\mathbb{N} \vDash_{\Sigma_n} \ulcorner \phi \urcorner [v_{\ulcorner x \urcorner}^0] \wedge \bigwedge m(\mathbb{N} \vDash_{\Sigma_n} \ulcorner \phi \urcorner [v_{\ulcorner x \urcorner}^m] \to \mathbb{N} \vDash_{\Sigma_n} \ulcorner \phi \urcorner [v_{\ulcorner x \urcorner}^{m+1}])$$

$$\to \bigwedge m\ \mathbb{N} \vDash_{\Sigma_n} \ulcorner \phi \urcorner [v_{\ulcorner x \urcorner}^m],$$

lo cual por (6.2) equivale a su vez a

$$\phi(0) \wedge \bigwedge m(\phi(m) \to \phi(m+1)) \to \bigwedge m\, \phi(m),$$

que es el principio de inducción para ϕ. ∎

No hemos sido capaces de definir en $\mathrm{I}\Sigma_1$ una (única) fórmula $\mathbb{N} \vDash \phi[v]$ con dos variables libres ϕ y v, sino infinitas fórmulas $\mathbb{N} \vDash_{\Sigma_n} \phi[v]$ con dos variables libres ϕ y v, donde la n no es una tercera variable, sino una metavariable que determina una de las infinitas fórmulas

$$\mathbb{N} \vDash_{\Sigma_0} \phi[v], \quad \mathbb{N} \vDash_{\Sigma_1} \phi[v], \quad \mathbb{N} \vDash_{\Sigma_2} \phi[v], \quad \mathbb{N} \vDash_{\Sigma_3} \phi[v], \quad \mathbb{N} \vDash_{\Sigma_4} \phi[v], \ \ldots$$

de tipos Δ_1, Σ_1, Σ_2, Σ_3, Σ_4, etc. (cada una de longitud mayor que la anterior).

Cada una de estas fórmulas nos permite formalizar todo "para toda fórmula Σ_n" metamatemática y reducir así un esquema axiomático o teoremático a un único axioma o teorema. El teorema anterior es un ejemplo muy claro de cómo $\mathbb{N} \models_{\Sigma_n} \phi[v]$ nos ha permitido reducir el esquema axiomático de Σ_n-inducción a un único axioma (acompañado de los axiomas necesarios para demostrar las propiedades de la fórmula $\mathbb{N} \models_{\Sigma_n} \phi[v]$).

6.5 Teoremas de corrección

El sueño de todo matemático sería que todas las fórmulas verdaderas fueran demostrables, pero esto no es cierto. Una afirmación menos ambiciosa es que todo lo demostrable sea verdadero (es decir, que el cálculo deductivo es correcto). En nuestro contexto, la forma más fiel en que podemos plasmar esta conjetura es ésta:

$$\bigwedge \alpha \in \Sigma_n (\underset{\ulcorner \mathrm{I}\Sigma_n \urcorner}{\vdash} \alpha \to \mathbb{N} \models_{\Sigma_n} \alpha),$$

donde $\mathbb{N} \models_{\Sigma_n} \alpha$ significa que $\mathbb{N} \models_{\Sigma_n} \alpha[v]$ se cumple para toda valoración v que cumpla $\mathrm{Val}(v, \alpha)$.

Esto puede demostrarse en $\mathrm{I}\Sigma_{n+1}$, pero la prueba no es trivial debido a un problema técnico: La forma natural de demostrar este teorema sería considerar una demostración de α e ir probando inductivamente que cada una de sus líneas cumple $\mathbb{N} \models_{\Sigma_n} \alpha$, pero esto no es viable porque no podemos garantizar que todas las líneas de la demostración sean de tipo Σ_n. Sin embargo, esto puede exigirse sin pérdida de generalidad si consideramos una demostración de α en términos del cálculo secuencial (teorema [CS 1.18]) y en esas condiciones el argumento natural funciona sin inconveniente alguno. Esto es el teorema [CS 1.25], cuyo enunciado transcribimos aquí:

Teorema 6.20 $\underset{\mathrm{I}\Sigma_{n+1}}{\vdash} \bigwedge \alpha \in \Sigma_n (\underset{\ulcorner \mathrm{I}\Sigma_n \urcorner}{\vdash} \alpha \to \mathbb{N} \models_{\Sigma_n} \alpha)$.

Combinando este teorema con 6.15 obtenemos:

Teorema 6.21 *Si α es una sentencia de tipo Σ_n en $\mathcal{L}_a$, entonces*

$$\underset{\mathrm{I}\Sigma_{n+1}}{\vdash} (\underset{\ulcorner \mathrm{I}\Sigma_n \urcorner}{\vdash} \ulcorner \alpha \urcorner \to \alpha).$$

En general, si T es una teoría que interpreta a $\mathcal{L}_a$, definimos

$$\mathrm{Consis}\, T \equiv \neg \underset{T}{\vdash} \ulcorner 0 \neq 0 \urcorner,$$

que es una sentencia que se interpreta como que T es consistente (recordemos que una teoría axiomática es contradictoria si en ella puede probarse una contradicción, lo cual equivale a que pueda demostrarse cualquier fórmula, por lo que la consistencia equivale a que una fórmula cualquiera, como $0 \neq 0$, no sea demostrable).

Si aplicamos el teorema precedente a la sentencia $0 \neq 0$ obtenemos que

$$\underset{\mathrm{I\Sigma_{n+1}}}{\vdash} (\underset{\ulcorner \mathrm{I\Sigma_n} \urcorner}{\vdash} \ulcorner 0 \neq 0 \urcorner \to 0 \neq 0),$$

luego:

Teorema 6.22 *Para cada número natural n se cumple*

$$\underset{\mathrm{I\Sigma_{n+1}}}{\vdash} \mathrm{Consis}\, \ulcorner \mathrm{I\Sigma_n} \urcorner.$$

En particular, en AP se demuestra la consistencia de todas las teorías $\mathrm{I}\Sigma_n$. Más en general, entendiendo que

$$\mathrm{Consis}\, \Gamma \equiv \neg\Gamma \vdash \ulcorner 0 = 0 \urcorner,$$

Teorema 6.23 (Teorema de reflexión) *Si Γ es cualquier conjunto finito de sentencias demostrables en* AP, *entonces* $\underset{\mathrm{AP}}{\vdash} \mathrm{Consis}\, \ulcorner \Gamma \urcorner$.

Demostración: Como cada sentencia de Γ es demostrable en $\mathrm{I}\Sigma_n$, para un n suficientemente grande, podemos tomar un mismo n que valga para todas las sentencias de Γ. Es claro entonces que

$$\underset{\mathrm{AP}}{\vdash} \mathrm{Consis}\, \ulcorner \mathrm{I\Sigma_n} \urcorner \to \mathrm{Consis}\, \ulcorner \Gamma \urcorner.$$

y basta aplicar el teorema anterior. ∎

Nota Quizá el lector deduzca de aquí que $\underset{\mathrm{AP}}{\vdash} \mathrm{Consis}\, \ulcorner \mathrm{AP} \urcorner$ con el argumento siguiente:

> *Si $\ulcorner \mathrm{AP} \urcorner$ fuera contradictorio, tendríamos $\underset{\ulcorner \mathrm{AP} \urcorner}{\vdash} 0 \neq 0$, pero la demostración requerirá únicamente un conjunto finito Γ de axiomas de $\ulcorner \mathrm{AP} \urcorner$, luego tendremos también $\Gamma \vdash 0 \neq 0$ y esto implica $\neg \mathrm{Consis}\, \Gamma$, en contradicción con el teorema de reflexión, luego $\ulcorner \mathrm{AP} \urcorner$ es consistente.*

Sin embargo, el "razonamiento" anterior es incorrecto o, más precisamente, no es formalizable en AP. Si, razonando en AP, suponemos que $\ulcorner \mathrm{AP} \urcorner$ es contradictorio, llegamos efectivamente a que existe un conjunto finito Γ de axiomas de $\ulcorner \mathrm{AP} \urcorner$ tal que $\neg\, \mathrm{Consis}\, \Gamma$, pero no podemos aplicar el teorema de reflexión. Éste sólo dice que si Γ es un conjunto finito (metamatemático) de axiomas de AP, entonces en AP se puede demostrar que $\ulcorner \Gamma \urcorner$ es consistente. Se trata de un esquema teoremático: para cada conjunto finito Γ tenemos una demostración de que $\ulcorner \Gamma \urcorner$ es consistente, pero eso no significa que

$$\underset{\mathrm{AP}}{\vdash} \bigwedge \Gamma (\Gamma \subset \mathrm{Ax}(\ulcorner \mathrm{AP} \urcorner) \to \mathrm{Consis}\, \Gamma).$$

Este presunto "teorema" sí que implicaría la consistencia de $\ulcorner \mathrm{AP} \urcorner$, pero nuestro razonamiento sólo nos da una variable Γ de la que sabemos que es un conjunto finito de axiomas de $\ulcorner \mathrm{AP} \urcorner$, no nos da un conjunto concreto $\ulcorner \Gamma \urcorner$ de axiomas de $\ulcorner \mathrm{AP} \urcorner$ que podamos probar que es consistente, así que no podemos concluir nada.

∎

Vamos a probar un teorema de corrección para Q análogo a 6.20, pero para ello necesitamos un resultado previo:

Teorema 6.24 *Toda sentencia de $\mathcal{L}_a$ de tipo Δ_0 es equivalente en* Q *a una sentencia sin cuantificadores (y, por lo tanto, sin variables).*

DEMOSTRACIÓN: Esto se prueba con un argumento inductivo muy simple. Lo que no es inmediato es que sea formalizable en $\mathrm{I}\Sigma_1$. Veamos primero el argumento y luego discutiremos su formalización. Vamos a asociar a cada sentencia α en $\mathcal{L}_a$ de tipo Δ_0 otra $\bar\alpha$ equivalente en Q y sin cuantificadores.

1. Si α es atómica basta tomar $\bar\alpha \equiv \alpha$.
2. Si $\alpha \equiv \neg\beta$, basta tomar $\bar\alpha \equiv \neg\bar\beta$.
3. Si $\alpha \equiv \beta \vee \gamma$, basta tomar $\bar\alpha \equiv \bar\beta \vee \bar\gamma$.
4. Si $\alpha \equiv \bigwedge u \le t\, \beta$, usando las dos primeras propiedades de 6.2 (junto con que $(0^{(n)})' = 0^{(n+\mathbf{1})}$), es fácil ver que, dado un designador t, existe un número natural n tal que $\vdash_Q t = 0^{(n)}$.

   Por lo tanto, α es equivalente a $\bigwedge u \le 0^{(n)}\, \beta(u)$ y, por 6.4, esto equivale a su vez a
   $$\beta(0^{(\mathbf{0})}) \wedge \beta(0^{(\mathbf{1})}) \wedge \cdots \wedge \beta(0^{(n)}),$$
   que a su vez equivale a la sentencia sin cuantificadores
   $$\bar\alpha \equiv \overline{\beta(0^{(\mathbf{0})})} \wedge \overline{\beta(0^{(\mathbf{1})})} \wedge \cdots \wedge \overline{\beta(0^{(n)})}.$$
5. Si $\alpha \equiv \bigvee u \le t\, \beta$, llegamos análogamente a
   $$\bar\alpha \equiv \overline{\beta(0^{(\mathbf{0})})} \vee \overline{\beta(0^{(\mathbf{1})})} \vee \cdots \vee \overline{\beta(0^{(n)})}.$$

Si la fórmula que probamos por inducción no tuviera que ser de tipo Σ_1, bastaría razonar por inducción sobre el número de signos lógicos de α (conectores y cuantificadores), pues en todos los casos aplicamos la hipótesis de inducción a una sentencia con menos signos lógicos. Para razonar en $\mathrm{I}\Sigma_1$ consideramos los números $c(v_0, \alpha_0, i)$ definidos al principio de la demostración de 6.11, de modo que si α_0 es una sentencia de tipo Δ_0 en $\mathcal{L}_a$ cuyas variables ligadas son $u_1, \ldots, u_n$, y aparecen acotadas en la forma $\bigwedge u_i \le t_i$ o $\bigvee u_i \le t_i$, y llamamos $V(\alpha_0)$ al conjunto de todas las valoraciones v definidas sobre las variables $u_1, \ldots, u_n$ tales que $v(u_i) \le c(\varnothing, \alpha_0, i)$, se cumple que $\mathrm{Dn}(t_i, v) \le c(\varnothing, \alpha_0, i)$. Más aún, si llamamos $c(\alpha_0)$ al máximo de los $c(\varnothing, \alpha_0, i)$, tenemos simplemente que $\mathrm{Dn}(t_i, v) \le c(\alpha_0)$.

Esto nos permite considerar los conjuntos (finitos) $F_j(\alpha_0)$ de todas las sentencias que resultan de sustituir las variables libres de una subsemifórmula β de α_0 con a lo sumo j signos lógicos por numerales $0^{(n)}$ con $n \le c(\alpha_0)$. Así, si α_0 tiene m signos lógicos, tenemos que $\alpha_0 \in F_m(\alpha_0)$, y podemos razonar por inducción sobre m que toda sentencia $\beta \in F_j(\alpha_0)$ es equivalente en Q a una sentencia sin cuantificadores.

Observemos que la fórmula $\alpha \in F_j(\alpha_0)$ es Δ_0 en $\mathrm{I}\Sigma_1$ (porque puede definirse en ARP) y, fijando $x = F_m(\alpha_0)$, la fórmula a la que le aplicamos la inducción dice que "para todo $\alpha \in x$" (cuantificador acotado) "si $\alpha \in F_j(\alpha_0)$, existe una sentencia $\bar{\alpha}$ sin cuantificadores y una demostración d en Q de $\alpha \leftrightarrow \bar{\alpha}$", con lo que se trata de una fórmula de tipo Σ_1 en $\mathrm{I}\Sigma_1$ y la inducción es formalizable.

Sólo tenemos que observar que, en los casos correspondientes a los cuantificadores acotados, si $\alpha \in F_j(\alpha_0)$, tenemos que $n = \mathrm{Dn}(0^{(n)}) = \mathrm{Dn}(t)$, donde el término t resulta de sustituir variables por numerales en alguno de los términos t_i que acotan variables en α_0, luego $n = \mathrm{Dn}(t) = \mathrm{Dn}(t_i, v)$, para cierta valoración $v \in V(\alpha_0)$, luego $n \leq c(\alpha_0)$, luego las sentencias $\beta(0^{(i)})$ consideradas en la prueba están en $F_k(\alpha_0)$, para un $k < j$, luego se les puede aplicar la hipótesis de inducción. ∎

Nota Observemos que, en la prueba del teorema anterior, a la vez que se demuestra inductivamente que $\alpha \leftrightarrow \bar{\alpha}$ es demostrable en Q, podemos comprobar también que $\mathbb{N} \vDash_0 \alpha \leftrightarrow \mathbb{N} \vDash_0 \bar{\alpha}$, que es una fórmula Δ_1 en $\mathrm{I}\Sigma_1$, luego la fórmula a la que le aplicamos el principio de inducción no deja de ser de tipo Σ_1. ∎

Esto es lo que necesitamos para probar el teorema [CS 1.26]:

Teorema 6.25 $\underset{\mathrm{I}\Sigma_1}{\vdash} \bigwedge \alpha \in \Delta_0 (\underset{\ulcorner \mathrm{Q} \urcorner}{\vdash} \alpha \to \mathbb{N} \vDash_0 \alpha)$.

Y, combinando este teorema con 6.15, obtenemos

Teorema 6.26 *Si α es una sentencia de tipo Δ_0 en $\mathcal{L}_a$, entonces*

$$\underset{\mathrm{I}\Sigma_1}{\vdash} (\underset{\ulcorner \mathrm{Q} \urcorner}{\vdash} \ulcorner \alpha \urcorner \to \alpha).$$

En particular:

Teorema 6.27 $\underset{\mathrm{I}\Sigma_1}{\vdash} \mathrm{Consis}\, \ulcorner \mathrm{Q} \urcorner$.

Ahora podemos probar una variante del teorema de Σ_1-completitud de Q para fórmulas de tipo Δ_1 en $\mathrm{I}\Sigma_1$. En principio, si α es una fórmula de este tipo, es decir, una fórmula equivalente en $\mathrm{I}\Sigma_1$ a una fórmula de tipo Σ_1 y a otra de tipo Π_1, podríamos aplicar el teorema 6.18 a la primera y a la negación de la segunda, pero el resultado que obtenemos es débil, porque ambas fórmulas no tienen por qué ser equivalentes en Q. Sin embargo, podemos afinar el argumento:

Teorema 6.28 *Si $\alpha(x_1, \ldots, x_n)$ es una fórmula de tipo Δ_1 en $\mathrm{I}\Sigma_1$ cuyas variables libres estén entre las indicadas, existe una fórmula $\phi(x_1, \ldots, x_n)$ de tipo Σ_1 equivalente a α en $\mathrm{I}\Sigma_1$ tal que*

$$\underset{\mathrm{I}\Sigma_1}{\vdash} (\phi(x_1, \ldots, x_n) \leftrightarrow \underset{\ulcorner Q \urcorner}{\vdash} \ulcorner \phi \urcorner (0^{(x_1)}, \ldots, 0^{(x_n)})).$$

$$\underset{\mathrm{I}\Sigma_1}{\vdash} (\neg\phi(x_1, \ldots, x_n) \leftrightarrow \underset{\ulcorner Q \urcorner}{\vdash} \ulcorner \neg\phi \urcorner (0^{(x_1)}, \ldots, 0^{(x_n)})).$$

DEMOSTRACIÓN: Pongamos que α equivale en $\mathrm{I}\Sigma_1$ a las fórmulas

$$\bigvee u\, \sigma(u, x_1, \ldots, x_n), \qquad \bigwedge v\, \tau(v, x_1, \ldots, x_n),$$

donde σ y τ son fórmulas de tipo Δ_0. Basta considerar

$$\phi \equiv \bigvee u\, (\sigma(u, x_1, \ldots, x_n) \wedge \bigwedge v \le u\, \tau(v, x_1, \ldots, x_n)).$$

Ciertamente es una fórmula de tipo Σ_1, y es equivalente a α, pues si se cumple $\alpha(x_1, \ldots, x_n)$, entonces tenemos que

$$\bigvee u\, \sigma(u, x_1, \ldots, x_n) \wedge \bigwedge v\, \tau(v, x_1, \ldots, x_n),$$

y esto implica claramente $\phi(x_1, \ldots, x_n)$. Recíprocamente, si $\phi(x_1, \ldots, x_n)$, trivialmente tenemos $\bigvee u\, \sigma(u, x_1, \ldots, x_n)$, luego se cumple $\alpha(x_1, \ldots, x_n)$.

Si suponemos $\phi(x_1, \ldots, x_n)$, el teorema 6.18 nos da $\underset{\ulcorner Q \urcorner}{\vdash} \ulcorner \phi \urcorner(0^{(x_1)}, \ldots, 0^{(x_n)})$.

Por otra parte, si se cumple $\neg\phi(x_1, \ldots, x_n)$, existe un número m tal que $\neg\tau(m, x_1, \ldots, x_n)$, luego 6.18 implica $\underset{\ulcorner Q \urcorner}{\vdash} \ulcorner \neg\tau \urcorner(0^{(m)}, 0^{(x_1)}, \ldots, 0^{(x_n)})$. Vamos a razonar en $\ulcorner \mathrm{Q} \urcorner$ que se cumple $\ulcorner \neg\phi \urcorner(0^{(x_1)}, \ldots, 0^{(x_n)})$. Por reducción al absurdo, suponemos que existe un u tal que

$$\ulcorner \sigma \urcorner(u, 0^{(x_1)}, \ldots, 0^{(x_n)}) \wedge \bigwedge v \le u\, \ulcorner \tau \urcorner(v, 0^{(x_1)}, \ldots, 0^{(x_n)}).$$

Por 6.4 tenemos que $u \le 0^{(m)} \vee 0^{(m)} \le u$, pero el segundo caso es imposible, ya que implica $\ulcorner \tau \urcorner(0^{(m)}, 0^{(x_1)}, \ldots, 0^{(x_n)})$, y podemos probar lo contrario. Así pues, tiene que ser $u \le 0^{(m)}$, y de nuevo por 6.4 tenemos:

$$u = 0^{(\mathbf{0})} \vee u = 0^{(\mathbf{1})} \vee \cdots \vee u = 0^{(m)},$$

luego

$$\ulcorner \sigma \urcorner(0, 0^{(x_1)}, \ldots, 0^{(x_n)}) \vee \cdots \vee \ulcorner \sigma \urcorner(0^{(m)}, 0^{(x_1)}, \ldots, 0^{(x_n)}).$$

Pero de $\neg\phi(x_1, \ldots, x_n)$ se sigue que todo u cumple $\neg\sigma(u, x_1, \ldots, x_n)$, luego 6.18 nos da que, para todo $i \le m$, se cumple

$$\underset{\ulcorner Q \urcorner}{\vdash} \ulcorner \neg\sigma \urcorner(0^{(i)}, 0^{(x_1)}, \ldots, 0^{(x_n)}),$$

luego en $\ulcorner \mathrm{Q} \urcorner$ podemos demostrar

$$\ulcorner \neg\sigma \urcorner(0, 0^{(x_1)}, \ldots, 0^{(x_n)}) \wedge \cdots \wedge \ulcorner \neg\sigma \urcorner(0^{(m)}, 0^{(x_1)}, \ldots, 0^{(x_n)})$$

y tenemos una contradicción que prueba $\ulcorner \neg\phi \urcorner(0^{(x_1)}, \ldots, 0^{(x_n)})$.

Esto prueba las dos implicaciones $\to$ del enunciado. Las opuestas son consecuencia de la consistencia de $\ulcorner \mathrm{Q} \urcorner$ (que es demostrable en $\mathrm{I}\Sigma_1$). Por ejemplo, si se cumple $\underset{\ulcorner Q \urcorner}{\vdash} \ulcorner \phi \urcorner(0^{(x_1)}, \ldots, 0^{(x_n)})$, tiene que ser $\phi(x_1, \ldots, x_n)$, ya que si se cumpliera $\neg\phi(x_1, \ldots, x_n)$, por la parte ya probada, tendríamos

$$\underset{\ulcorner Q \urcorner}{\vdash} \ulcorner \neg\phi \urcorner(0^{(x_1)}, \ldots, 0^{(x_n)}),$$

y entonces $\ulcorner \mathrm{Q} \urcorner$ sería contradictorio. ∎

Nota Observando la prueba del teorema anterior vemos que vale igualmente si reemplazamos $\mathrm{I}\Sigma_1$ por cualquier extensión suya T (de modo que la fórmula α es Δ_1^T, en cuyo caso ϕ es equivalente a α en T). ∎

Usaremos también esta variante del teorema anterior:

Teorema 6.29 *Sea $\alpha(x_1,\ldots,x_n,y)$ una fórmula de $\mathcal{L}_a$ de tipo Σ_1 cuyas variables libres estén entre las indicadas y tal que*

$$\underset{\mathrm{I}\Sigma_1}{\vdash} \bigwedge uv(\alpha(x_1,\ldots,x_n,u)\wedge\alpha(x_1,\ldots,x_n,v)\to u=v).$$

Llamemos $F(x_1,\ldots,x_n)\equiv y|\alpha(x_1,\ldots,x_n,y)$. Entonces existe una fórmula $\phi(x_1,\ldots,x_n,y)$ de tipo Σ_1 equivalente a α en $\mathrm{I}\Sigma_1$ tal que

$$\underset{\mathrm{I}\Sigma_1}{\vdash}\Big(\bigvee y\,\alpha(x_1,\ldots,x_n,y)\to \underset{\ulcorner Q\urcorner}{\vdash}\bigwedge y(\ulcorner\phi\urcorner(0^{(x_1)},\ldots,0^{(x_n)},y)\leftrightarrow y=0^{(F(x_1,\ldots,x_n))})\Big).$$

Demostración: Por abreviar la notación supondremos que $n=1$, es decir, escribiremos x en lugar de $x_1,\ldots,x_n$, pero es claro que la prueba vale para cualquier número de argumentos sin cambio alguno.

Pongamos que $\alpha(x,y)\equiv\bigvee z\,\sigma(x,y,z)$, donde σ es de tipo Δ_0. Definimos

$$\tau(x,y,z)\equiv\sigma(x,y,z)\wedge\bigwedge uv\le y(u\ne y\to\neg\sigma(x,u,v))$$
$$\wedge\bigwedge uv\le z(u\ne y\to\neg\sigma(x,u,v)),$$

que es una fórmula de tipo Δ_0, por lo que $\phi(x,y)\equiv\bigvee z\,\tau(x,y,z)$ es de tipo Σ_1. Veamos que $\phi(x,y)\leftrightarrow\alpha(x,y)$.

Si se cumple $\alpha(x,y)$, existe un z tal que $\sigma(x,y,z)$, pero, por la hipótesis de unicidad, para cualquier $u\ne y$ y cualquier v se cumple $\neg\alpha(x,u)$, luego $\neg\sigma(x,u,v)$, y esto implica $\tau(x,y,z)$, luego $\phi(x,y)$. Recíprocamente, si se cumple $\phi(x,y)$, existe un z tal que $\tau(x,y,z)$, luego $\sigma(x,y,z)$, luego $\alpha(x,y)$.

Ahora supongamos $\bigvee y\,\alpha(x,y)$, con lo que también existe un $y=F(x)$ tal que $\phi(x,y)$, luego existe un z tal que $\sigma(x,y,z)$. Por el teorema de Σ_1-completitud de Q (teorema 6.18), concluimos que

$$\underset{\ulcorner Q\urcorner}{\vdash}\ulcorner\sigma\urcorner(0^{(x)},0^{(y)},0^{(z)}),$$

luego

$$\underset{\ulcorner Q\urcorner}{\vdash}\ulcorner\phi\urcorner(0^{(x)},0^{(y)}),$$

luego

$$\underset{\ulcorner Q\urcorner}{\vdash}\bigwedge y(y=0^{(F(x))}\to\ulcorner\phi\urcorner(0^{(x)},y)).$$

Para probar la implicación contraria razonamos en $\ulcorner\mathrm{Q}\urcorner$. Seguimos llamando $x,y=F(x),z$ a los tres números que hemos fijado, y ahora suponemos que se cumple $\ulcorner\phi\urcorner(0^{(x)},y_0)$, lo cual significa que existe un z_0 tal que $\ulcorner\tau\urcorner(0^{(x)},y_0,z_0)$. Tenemos que probar que $y_0=0^{(y)}$, así que, por reducción al absurdo, suponemos $y_0\ne 0^{(y)}$.

Sea $h = \text{máx}\{y, z\}$. Por 6.4 tenemos que

$$(0^{(h)} \leq y_0 \vee y_0 \leq 0^{(h)}) \wedge (0^{(h)} \leq z_0 \vee z_0 \leq 0^{(h)}).$$

Supongamos en primer lugar que $0^{(h)} \leq y_0 \vee 0^{(h)} \leq z_0$. Entonces, o bien $0^{(y)} \leq y_0 \wedge 0^{(z)} \leq y_0$ o bien $0^{(y)} \leq z_0 \wedge 0^{(z)} \leq z_0$. En cualquiera de los dos casos $\ulcorner\tau\urcorner(0^{(x)}, y_0, z_0)$ implica $\neg\ulcorner\sigma\urcorner(0^{(x)}, 0^{(y)}, 0^{(z)})$, lo que nos da una contradicción, pues en Q puede probarse $\ulcorner\sigma\urcorner(0^{(x)}, 0^{(y)}, 0^{(z)})$.

Por consiguiente tiene que ser $y_0 \leq 0^{(h)} \wedge z_0 \leq 0^{(h)}$, luego, o bien tenemos $y_0 \leq 0^{(y)} \wedge z_0 \leq 0^{(y)}$, o bien $y_0 \leq 0^{(z)} \wedge z_0 \leq 0^{(z)}$. Como en Q se demuestra $\ulcorner\tau\urcorner(0^{(x)}, 0^{(y)}, 0^{(z)})$, tenemos $\neg\ulcorner\sigma\urcorner(0^{(x)}, y_0, z_0)$, cuando estamos suponiendo lo contrario. ∎

6.6 Teorías semirrecursivas

En la sección 3.2 introdujimos los lenguajes formales de primer orden y explicamos cómo se puede formalizar la definición en (la versión original de) ARP. Si queremos considerar esta definición en ARP como teoría de primer orden, o en $\mathrm{I}\Sigma_1$, conviene introducir una precisión adicional que no tenía sentido hacer en aquel contexto.

En principio, podríamos definir un lenguaje formal en $\mathrm{I}\Sigma_1$ fijando, entre otras cosas, una fórmula $x \in \mathrm{VarLib}(\ulcorner\mathcal{L}\urcorner)$ que cumpliera todos los requisitos de la definición, en particular que hay infinitos números naturales que la cumplen, pero que fuera tan complicada que no pudéramos saber si un número natural es o no una variable libre de $\ulcorner\mathcal{L}\urcorner$. A su vez, esto haría que no pudiéramos asegurar si una cadena de signos dada es o no una fórmula de $\ulcorner\mathcal{L}\urcorner$, etc.

Esta situación no tiene ningún interés. Queremos los lenguajes formales para hablar sin trabas. Una cosa es que no sepamos si una fórmula es verdadera o falsa, o si es demostrable o no, y otra muy distinta que no sepamos si una sucesión de números naturales es una fórmula o no. Por ello vamos a considerar únicamente lenguajes formales definibles en (la versión original de) ARP, lo cual equivale a que estén definidos por fórmulas atómicas en $\mathrm{I}\Sigma_1$ (dotado de los funtores de $\mathcal{L}_{\mathrm{arp}}$ o, en virtud del teorema 4.50:

> *Vamos a considerar únicamente lenguajes formales definidos en* $\mathrm{I}\Sigma_1$ *mediante fórmulas* $\Delta_1^{\mathrm{I}\Sigma_1}$.

Esto se expresa también diciendo que consideramos únicamente lenguajes formales $\mathcal{L}$ *demostrablemente recursivos,* en alusión a que en $\mathrm{I}\Sigma_1$ se tiene que poder demostrar que cada fórmula que forma parte de la definición de un lenguaje formal es equivalente en $\mathrm{I}\Sigma_1$ a una fórmula Σ_1 y a otra Π_1.

Como hemos señalado, esto equivale a que el lenguaje formal $\mathcal{L}$ es definible en (la versión original de) ARP, lo cual se traduce en que la fórmula

$$\Gamma \overset{d}{\vdash} \alpha$$

(con tres variables libres) que expresa que d es una deducción de la fórmula α de $\mathcal{L}$ a partir del conjunto (finito) de premisas Γ es atómica en $\mathcal{L}_{\mathrm{arp}}$ y es $\Delta_1^{\mathrm{I}\Sigma_1}$ en $\mathcal{L}_a$. En cambio, de la fórmula

$$\Gamma \vdash \alpha \equiv \bigvee d\, \Gamma \overset{d}{\vdash} \alpha$$

sólo podemos asegurar que es Σ_1.

Para definir una teoría axiomática de primer orden T en $\mathrm{I}\Sigma_1$ sobre un lenguaje formal $\mathcal{L}$ sólo es necesario especificar una fórmula $\alpha \in \mathrm{Ax}(T)$ que determine qué fórmulas de $\mathcal{L}$ son axiomas (propios) de T.

Ahora no podemos afirmar que en algunos contextos no tenga interés considerar teorías axiomáticas tales que no exista un criterio explícito para determinar si una fórmula dada es o no un axioma, pero habitualmente será razonable exigirlo, así que introducimos las definiciones siguientes:

Definición 6.30 Una teoría axiomática de primer orden T definida en $\mathrm{I}\Sigma_1$ es *recursiva* (resp. *semirrecursiva*) si la fórmula $\alpha \in \mathrm{Ax}(T)$ que define los axiomas de T es de tipo $\Delta_1^{\mathrm{I}\Sigma_1}$ (resp. de tipo Σ_1.)

Si T es una teoría axiomática recursiva, entonces T puede definirse en (la versión original de) ARP, por lo que la fórmula

$$\Gamma \underset{T}{\overset{d}{\vdash}} \alpha$$

es atómica en ARP, luego es $\Delta_1^{\mathrm{I}\Sigma_1}$, luego $\Gamma \underset{T}{\vdash} \alpha$ es Σ_1. Ahora bien, si T es meramente semirrecursiva llegamos a la misma conclusión, pues

$$\Gamma \underset{T}{\vdash} \alpha \leftrightarrow \bigvee d\Delta(\bigwedge \alpha \in \Delta\, (\alpha \in \Gamma \vee \alpha \in \mathrm{Ax}(T)) \wedge \Delta \overset{d}{\vdash} \alpha),$$

y podemos tomar como definición de $\Gamma \underset{T}{\vdash} \alpha$ una fórmula de tipo Σ_1 en sentido estricto equivalente a la fórmula de la derecha. Muchos resultados válidos para teorías axiomáticas recursivas valen igualmente para teorías semirrecursivas debido al teorema siguiente:

Teorema 6.31 (Craig) *Si T es una teoría axiomática semirrecursiva, existe una teoría axiomática recursiva con los mismos teoremas.*

Demostración: Sea $\alpha \in \mathrm{Ax}(\ulcorner T \urcorner) \equiv \bigvee u\, \phi(\alpha, u)$, donde la fórmula ϕ es de tipo Δ_0. Sea

$$\alpha^{(n)} = \bigwedge_{i=1}^{n} \alpha.$$

Consideramos la teoría T' determinada por la fórmula Δ_1:

$$\beta \in \mathrm{Ax}(T') \equiv \bigvee u n \alpha \leq \beta (\beta = \alpha^{(n)} \wedge \phi(\alpha, u))$$

(notemos que la fórmula $\beta = \alpha^{(n)}$ es Δ_1). Así T' es una teoría axiomática recursiva y tiene los mismos teoremas que T, pues si β es un axioma de T', entonces existen u, n, α tales que $\beta = \alpha^{(n)}$ y $\phi(\alpha, u)$, luego $\bigvee u\, \phi(\alpha, u)$, lo que significa que α es un axioma de T, y es claro entonces que $\underset{T}{\vdash} \beta$.

Recíprocamente, si α es un axioma de T, existe un u tal que $\phi(\alpha, u)$, y es claro que, para un n suficientemente grande, la conjunción $\beta \equiv \alpha^{(n)}$, vista como número natural, es mayor que u y que n, luego $\beta \in \mathrm{Ax}(T')$, luego $\underset{T'}{\vdash} \alpha$.

Como todo axioma de T es un teorema de T' y viceversa, es claro que T y T' tienen los mismos teoremas. ∎

Ahora probaremos tres resultados fundamentales sobre la formalización del concepto de demostrabilidad:

Teorema 6.32 (Condiciones de Hilbert-Bernays) *Sea T una teoría axiomática semirrecursiva sobre un lenguaje formal $\mathcal{L}$ que interprete a* Q *y sean ϕ, ψ fórmulas de $\mathcal{L}$. Entonces:*

1. *Si $\underset{T}{\vdash} \phi$, entonces $\underset{\mathrm{Q}}{\vdash} \underset{\ulcorner T \urcorner}{\vdash} \ulcorner \phi \urcorner$,*
2. $\underset{\mathrm{I}\Sigma_1}{\vdash} (\underset{\ulcorner T \urcorner}{\vdash} \ulcorner \phi \urcorner \to \underset{\ulcorner T \urcorner}{\vdash} \ulcorner \underset{\ulcorner T \urcorner}{\vdash} \ulcorner \phi \urcorner \urcorner)$,
3. $\underset{\mathrm{I}\Sigma_1}{\vdash} (\underset{\ulcorner T \urcorner}{\vdash} \ulcorner \phi \urcorner \wedge \underset{\ulcorner T \urcorner}{\vdash} \ulcorner \phi \to \psi \urcorner \to \underset{\ulcorner T \urcorner}{\vdash} \ulcorner \psi \urcorner)$.

Demostración: La única dificultad de este teorema es entender exactamente el enunciado. Tal y como hemos explicado en la introducción de este capítulo, estamos trabajando en una metateoría informal $\mathrm{I}\Sigma_1$ (sobre un lenguaje $\mathcal{L}_a^0$) en la cual tenemos definida la teoría $\mathrm{I}\Sigma_1$ sobre un lenguaje $\mathcal{L}_a$. A su vez, en la teoría $\mathrm{I}\Sigma_1$ tenemos definida la formalización $\ulcorner \mathrm{I}\Sigma_1 \urcorner$ sobre el lenguaje formal $\ulcorner \mathcal{L}_a \urcorner$.

En el enunciado hay que entender que T es una teoría axiomática definida en la metateoría $\mathrm{I}\Sigma_1$, al igual que la aritmética de Robinson Q, pero podemos considerar también sus formalizaciones en la teoría $\mathrm{I}\Sigma_1$, a las que llamamos $\ulcorner T \urcorner$ y $\ulcorner \mathrm{Q} \urcorner$, respectivamente, sobre los lenguajes $\ulcorner \mathcal{L}_a \urcorner$ y $\ulcorner \mathcal{L} \urcorner$, respectivamente.

Según hemos visto, el hecho de que T sea semirrecursiva se traduce en que la fórmula $\underset{T}{\vdash} \phi$ de $\mathcal{L}_a^0$ es de tipo Σ_1, y podemos considerar su formalización $\ulcorner \underset{T}{\vdash} \phi \urcorner \equiv \underset{\ulcorner T \urcorner}{\vdash} \ulcorner \phi \urcorner$ en $\mathcal{L}_a$. Observemos que ϕ es una variable de $\mathcal{L}_a^0$ y $\ulcorner \phi \urcorner$ es una variable de $\mathcal{L}_a$.

Ahora, el teorema 6.18 nos da que (en la metateoría $\mathrm{I}\Sigma_1$) se demuestra

$$\underset{T}{\vdash} \phi \to \underset{Q}{\vdash} \underset{\ulcorner T \urcorner}{\vdash} 0^{(\phi)},$$

pero si $\phi \in \mathrm{Form}(\mathcal{L})$, el numeral $0^{(\phi)}$ de $\mathcal{L}_a$ es la formalización de ϕ, es decir, la fórmula de $\ulcorner \mathcal{L} \urcorner$ que usualmente representamos por $\ulcorner \phi \urcorner$, pero que no hay que confundir con la variable[12] $\ulcorner \phi \urcorner$ de $\mathcal{L}_a$. Con esto tenemos 1.

[12] Cuando $\ulcorner \phi \urcorner$ nombra una variable de $\mathcal{L}_a$, los ángulos de Quine son los que asignan a cada cadena de $\mathcal{L}_a^0$ una cadena de $\mathcal{L}_a$, mientras que cuando consideramos a $\ulcorner \phi \urcorner$ como fórmula del lenguaje $\ulcorner \mathcal{L} \urcorner$ nos referimos a los ángulos de Quine definidos en $\mathcal{L}_a$ que asignan a cada cadena de $\mathcal{L}$ una cadena de $\ulcorner \mathcal{L} \urcorner$.

Para 2. observamos que $\underset{\ulcorner T\urcorner}{\vdash} \ulcorner\phi\urcorner$ es una sentencia de $\mathcal{L}_a$ de tipo Σ_1, y ahora el teorema 6.18 nos da que

$$\underset{\mathrm{I}\Sigma_1}{\vdash} (\underset{\ulcorner T\urcorner}{\vdash} \ulcorner\phi\urcorner \to \underset{\ulcorner Q\urcorner}{\vdash} \ulcorner \underset{\ulcorner T\urcorner}{\vdash} \ulcorner\phi\urcorner\urcorner),$$

donde $\ulcorner \underset{\ulcorner T\urcorner}{\vdash} \ulcorner\phi\urcorner\urcorner$ es una fórmula de $\ulcorner\mathcal{L}_a\urcorner$.

Ahora bien, que T interpreta a Q significa que (en la metateoría $\mathrm{I}\Sigma_1$) podemos asociar a cada fórmula de $\mathcal{L}_a$ una traducción en $\mathcal{L}$, de modo que (teorema 5.23) las traducciones de los teoremas de Q son teoremas de T. En particular, en la metateoría $\mathrm{I}\Sigma_1$ se demuestra la fórmula

$$\underset{Q}{\vdash} \underset{\ulcorner T\urcorner}{\vdash} \ulcorner\phi\urcorner \to \underset{T}{\vdash} \underset{\ulcorner T\urcorner}{\vdash} \ulcorner\phi\urcorner,$$

donde la fórmula tras $\underset{T}{\vdash}$ ha de entenderse como la traducción a $\mathcal{L}$ de la fórmula correspondiente de $\mathcal{L}_a$.

Por la nota tras el teorema 6.18, si esta sentencia de $\mathcal{L}_a^0$ de tipo Σ_1 es demostrable en la metateoría $\mathrm{I}\Sigma_1$, también podemos demostrar en la teoría Q formalizada su formalización en $\mathcal{L}_a$, es decir, que

$$\underset{Q}{\vdash} (\underset{\ulcorner Q\urcorner}{\vdash} \ulcorner \underset{\ulcorner T\urcorner}{\vdash} \ulcorner\phi\urcorner\urcorner \to \underset{\ulcorner T\urcorner}{\vdash} \ulcorner \underset{\ulcorner T\urcorner}{\vdash} \ulcorner\phi\urcorner\urcorner),$$

donde $\ulcorner \underset{\ulcorner T\urcorner}{\vdash} \ulcorner\phi\urcorner\urcorner$ ha de entenderse primero como una fórmula de $\ulcorner\mathcal{L}_a\urcorner$ y luego como su traducción a $\ulcorner\mathcal{L}\urcorner$. En particular podemos cambiar $\underset{\mathrm{Q}}{\vdash}$ por $\underset{\mathrm{I}\Sigma_1}{\vdash}$ y, encadenando las dos implicaciones que hemos obtenido, llegamos a 2.

La propiedad 3. es trivial, pues sólo es la formalización en $\mathrm{I}\Sigma_1$ de la regla del Modus Ponens. ∎

Capítulo VII

Lógica de segundo orden

Una teoría aritmética como AP o $\mathrm{I}\Sigma_1$ está diseñada para hablar de números naturales, pero en ella podemos hablar de conjuntos finitos de números naturales gracias a que cada uno de ellos se puede identificar con un número natural. Sin embargo, en la práctica también podemos hablar indirectamente de conjuntos infinitos de números naturales, como el conjunto de todos los números primos, etc., a través de fórmulas, pues cada fórmula determina el conjunto (tal vez infinito) de todos los objetos para los que se cumple. Más en general, en cualquier teoría axiomática, las fórmulas con n variables nos permiten hablar de relaciones n-ádicas e incluso de funciones n-ádicas que asignan a cada n objetos de los que habla la teoría un objeto imagen.

Sin embargo, esta forma de estudiar formalmente conjuntos, relaciones y funciones a través de las fórmulas que los definen tiene sus limitaciones. Por ejemplo, en $\mathrm{I}\Sigma_1$ podemos considerar fórmulas de $\mathcal{L}_a$ que hablen del conjunto de los números primos, del conjunto de los múltiplos de 10, etc., pero no podemos construir una fórmula que diga "para todo conjunto" o "existe un conjunto", porque ninguna fórmula de $\mathcal{L}_a$ puede significar "para toda fórmula" o "existe una fórmula" (otra cosa es que podemos cuantificar sobre las fórmulas de la formalización $\ulcorner\mathcal{L}_a\urcorner$ de $\mathcal{L}_a$, pero no nos permite construir una fórmula de $\mathcal{L}_a$ que cuantifique sobre los conjuntos de los que podemos hablar en $\mathcal{L}_a$ a través de fórmulas de $\mathcal{L}_a$).

La situación es similar a la que nos encontrábamos con la versión original de ARP, en la que podíamos hablar de números naturales, pero no cuantificar sobre ellos. Sólo podíamos expresar que una afirmación es cierta para todos los números naturales demostrando que es así para un x arbitrario, y sólo podíamos demostrar que existe un x que cumple alguna condición definiéndolo explícitamente. Esto lo resolvimos pasando a la lógica de primer orden, lo que nos permitió añadir cuantificadores al lenguaje de ARP, de modo que "para todo número x" y "existe un número x" pasaron a ser expresables formalmente. Ahora vamos a hacer algo similar. Veremos que cualquier teoría axiomática de primer orden se puede extender a otra en la que es posible cuantificar sobre los conjuntos de objetos definidos por las fórmulas, así como sobre relaciones y

funciones n-ádicas. En realidad basta con considerar el caso de las relaciones, pues todo conjunto se puede ver como una relación monádica y toda función n-ádica se puede ver como una relación $n+1$-ádica.

Más precisamente, vamos a estudiar ahora lo que se conoce como lógica de segundo orden, es decir, teorías axiomáticas sobre lenguajes formales que incluyen variables de primer orden (que representan objetos) y variables de segundo orden (que representan relaciones n-ádicas entre objetos, en particular, conjuntos), de modo que los cuantificadores se puedan aplicar indistintamente a variables de primer o segundo orden.

En el capítulo II de [CS] definiremos lenguajes formales de segundo orden propiamente dichos, con variables de primer y segundo orden y axiomas y reglas de inferencia que regulan separadamente el uso de unas y otras. Sin embargo, aquí vamos a presentar la lógica de segundo orden con un enfoque diferente (que en [CS] probaremos que es equivalente) de modo que los lenguajes de segundo orden no serán más que una clase particular de lenguajes de primer orden, y las teorías de segundo orden serán teorías de primer orden con axiomas específicos que formalicen la distinción entre "objetos" y "relaciones entre objetos". Así podemos razonar en ellas usando el cálculo deductivo de primer orden que ya conocemos.

7.1 Teorías axiomáticas de segundo orden

Definición 7.1 Un *lenguaje formal de segundo orden* es un lenguaje formal de primer orden con igualador dotado de una sucesión de relatores $R_0, R_1, R_2, \ldots$ de rango 1 y otra sucesión de relatores $S_1, S_2, S_3 \ldots$ de modo que S_r tenga rango $r+1$. (Nos referiremos a estos relatores como *relatores estructurales.*) Usaremos la notación

$$t_0(t_1, \ldots, t_r) \equiv S_r(t_0, \ldots, t_r).$$

La idea subyacente es que R_0x pretende significar "x es un objeto", mientras que R_rX, para $r \geq 1$, pretende significar que "X es una relación de rango r" y $X(x_1, \ldots, x_r)$ que "los objetos $x_1, \ldots, x_r$ cumplen la relación X".

Decimos "pretende significar" porque, en principio, nada nos impide considerar fórmulas como

$$R_3X \wedge X(y, z),$$

que contradicen nuestra intención de que X sea en este caso una relación triádica y no diádica. Vamos a definir ahora las fórmulas de $\mathcal{L}$ que, además de cumplir los requisitos de la sintaxis de $\mathcal{L}$, son coherentes con el significado pretendido de los relatores adicionales.

Si $\mathcal{L}$ es un lenguaje de segundo orden, llamaremos variable libre de índice i y rango r a la variable $X_i^r \equiv X_{\langle r,i\rangle}$, mientras que la variable ligada de índice i y rango r será $U_i^r \equiv U_{\langle r,i\rangle}$, donde $\langle r, i\rangle$ es el par definido en 2.3. A las variables de rango 0 las llamaremos *variables de primer orden,* mientras que a las de rango no nulo las llamaremos *variables de segundo orden.*

La idea es que vamos a restringir el uso de las variables de modo que una variable X de rango r la usaremos únicamente para hacer referencia a objetos que cumplen R_rX. Con esta intención, definimos las abreviaturas:

$$\bigwedge_r U\,\alpha \equiv \bigwedge U(R_rU \to \alpha), \qquad \bigvee_r U\,\alpha \equiv \bigvee U(R_rU \wedge \alpha),$$

donde U es una variable de $\mathcal{L}$ de rango r.

Llamaremos *semitérminos estructurados* de $\mathcal{L}$ a los semitérminos sin descriptores que no contengan variables de segundo orden (a los que llamaremos *semitérminos estructurados de primer orden*) y a las variables de rango $r \geq 1$ (a las que llamaremos *semitérminos estructurados de rango r).*

Definimos las *semifórmulas estructuradas* como las semifórmulas de $\mathcal{L}$ determinadas por las reglas siguientes:

1. Si R^n es un relator n-ádico no estructural y $t_1, \ldots, t_n$ son semitérminos de primer orden, entonces $R^n(t_1, \ldots, t_n)$ es una semifórmula estructurada.
2. Si T es un semitérmino estructurado (una variable) de rango r y $t_1, \ldots, t_r$ son semitérminos estructurados de primer orden, entonces $T(t_1, \ldots, t_r)$ es una semifórmula estructurada.
3. Si α y β son semifórmulas estructuradas, también lo son

   $$\neg\alpha, \quad \alpha \vee \beta, \quad \bigwedge_r U\,\alpha, \quad \bigvee_r U\,\alpha,$$

   donde U es una variable ligada de rango r.

Claramente, si α y β son semifórmulas estructuradas, también lo son

$$\alpha \to \beta, \qquad \alpha \wedge \beta, \qquad \alpha \leftrightarrow \beta.$$

Las *semifórmulas estructuradas de primer orden* son aquellas cuyas variables ligadas son todas de primer orden (pero pueden tener variables libres de segundo orden).

Lenguajes reducidos En las teorías en las que es posible definir n-tuplas ordenadas no hay necesidad de considerar variables de segundo orden de todos los rangos, sino que es suficiente considerar variables de rango 1.

Un *lenguaje formal reducido de segundo orden* es un lenguaje formal con igualador dotado de un relator monádico R_1 y de un relator diádico S_1. Introducimos la notación:

$$R_0(t) \equiv \neg R_1(t), \qquad \mathrm{cto}\, t \equiv R_1(t), \qquad t \in T \equiv S_1(T, t),$$

La variable libre de primer orden de índice i se define como $x_i \equiv X_{2i}$, mientras que la variable libre de segundo orden de índice i es $X_i \equiv X_{2i+1}$. Igualmente se definen las variables ligadas $u_i \equiv U_{2i}$, $U_i \equiv U_{2i+1}$ de primer y segundo orden.

La definición de término estructurado y fórmula se adapta obviamente a este caso, en el que sólo tenemos términos de primer orden (de rango 0) y de segundo orden (de rango 1). ∎

Definición 7.2 Una *teoría axiomática de segundo orden* es una teoría axiomática de primer orden sobre un lenguaje formal de segundo orden entre cuyos axiomas propios se encuentren al menos los siguientes:

Axiomas estructurales

1. $\bigwedge_* u_1 \cdots u_n\, R_0 t$,
   para todo término estructurado[1] $t(x_1, \ldots, x_n)$ de primer orden,
2. $\bigvee_r U\, R_r U$, para $r \geq 0$,
3. $\bigwedge_r U\, \neg R_s U$, para $r \neq s$,

Extensionalidad (para todo $r \geq 1$):

$$\bigwedge_r UV(\bigwedge_0 u_1 \cdots u_r(U(u_1, \ldots, u_r) \leftrightarrow V(u_1, \ldots, u_r)) \rightarrow U = V).$$

En el caso de lenguajes con descriptor, supondremos que las teorías axiomáticas de segundo orden incluyen también el axioma $R_0(u|u = u)$.

Esta definición se adapta de forma obvia al caso de lenguajes reducidos.

Si $\mathcal{L}$ es un lenguaje formal de primer orden, podemos considerar un lenguaje de segundo orden (pleno o reducido) $\mathcal{L}^2$ que resulte de añadirle a $\mathcal{L}$ los relatores estructurales y nuevas variables, de modo que las variables de $\mathcal{L}$ sean las variables de primer orden de $\mathcal{L}^2$.

Podemos definir una interpretación de $K_\mathcal{L}$ (el el sentido de 5.19) en la teoría axiomática $K^2_\mathcal{L}$ que tiene por axiomas propios los axiomas estructurales y de extensionalidad que requiere la definición anterior (más $R_0(u|u = u)$ en el caso de lenguajes con descriptor). En efecto:

1. La traducción de cada variable de $\mathcal{L}$ es ella misma.
2. Como universo de la interpretación tomamos la fórmula $R_0 x$.
3. Si c es una constante de $\mathcal{L}$, su traducción es ella misma, y por los axiomas estructurales en $K^2_\mathcal{L}$ se demuestra que $R_0 c$.
4. Si f es un funtor n-ádico, su traducción es el término $f(x_1, \ldots, x_n)$ que, de nuevo por los axiomas estructurales, cumple lo requerido por la definición de interpretación.
5. Si R es un relator n-ádico de $\mathcal{L}$, su traducción es la fórmula $R(x_1, \ldots, x_n)$.

Así tenemos definida la traducción de cada semiexpresión de $\mathcal{L}$ a una semiespresión de $\mathcal{L}^2$, que se reduce a sustituir cada cuantificador $\bigwedge u$ o $\bigvee u$ por $\bigwedge_0 u$ y $\bigvee_0 u$, respectivamente y cada desciptor $u| \cdots$ por $u|(R_0 u \wedge \cdots)$.

Observemos que la traducción de toda fórmula sin descriptores es siempre una fórmula estructurada de $\mathcal{L}^2$.

[1] Las variables x_i son todas las variables libres en t, y pueden ser de cualquier rango. El asterisco en $\bigwedge_*$ indica que cada una se cuantifica según su rango.

Más en general, si T es cualquier teoría axiomática de primer orden sobre $\mathcal{L}$, podemos considerar la teoría axiomática de segundo orden T^2 sobre $\mathcal{L}^2$ cuyos axiomas son los dados por la definición 7.2 más las traducciones de las clausuras universales de los axiomas de T, y así sucede igualmente que T^2 interpreta a T.

Los resultados de [CS] nos permiten probar que T^2 es una extensión conservativa de T, en el sentido de que la traducción de una sentencia α de $\mathcal{L}$ es un teorema de T^2 si y sólo si α es un teorema de T. De hecho, vamos a demostrar que esto es así con teorías mucho más potentes que T^2, en la que apenas podemos probar obviedades sobre los objetos de segundo orden.

Definición 7.3 Si T es una teoría axiomática de primer orden sobre un lenguaje formal $\mathcal{L}$, llamamos *extensión predicativa* de T a la teoría axiomática de segundo orden sobre $\mathcal{L}^2$ cuyos axiomas son:

Axiomas estructurales

1. $\bigwedge_* u_1 \cdots u_n\, R_0 t$,
   para todo término estructurado[2] $t(x_1, \ldots, x_n)$ de primer orden,
2. $X(x_1, \ldots, x_r) \to R_r X \wedge R_0 x_1 \wedge \cdots \wedge R_0 x_r$
3. $\bigwedge_r U\, \neg R_s U$.

Extensionalidad (para todo $r \geq 1$):

$$\bigwedge_r UV(\bigwedge_0 u_1 \cdots u_r(U(u_1, \ldots, u_r) \leftrightarrow V(u_1, \ldots, u_r)) \to U = V).$$

Comprensión $\bigwedge \bar{V}_* \bigvee_r U \bigwedge_0 u_1 \cdots u_r(U(u_1, \ldots, u_r) \leftrightarrow \alpha(u_1, \ldots, u_r))$,
para toda fórmula estructurada de primer orden[3] $\alpha(x_1, \ldots, x_r)$.

Axiomas propios Las traducciones de las clausuras universales de los axiomas propios de T.

Si extendemos el axioma de comprensión a fórmulas arbitrarias (no necesariamente de primer orden) tenemos la *extensión plena* de T.

Notemos que hemos eliminado un axioma estructural porque se deduce del esquema de comprensión.

El resultado fundamental es el siguiente:

Teorema 7.4 *Si T es una teoría axiomática de primer orden sobre un lenguaje formal*[4] *$\mathcal{L}$, su extensión predicativa de segundo orden es una extensión conservativa, en el sentido de que una sentencia de $\mathcal{L}$ es un teorema de T si y sólo si su traducción es un teorema de la extensión predicativa.*

[2] Las variables x_i son todas las variables libres en t, y pueden ser de cualquier rango. El asterisco en $\bigwedge_*$ indica que cada una se cuantifica según su rango.

[3] Las variables $x_1, \ldots, x_n$ tienen rango 0. La fórmula α puede tener más variables libres de cualquier orden, pero todas ellas tienen que estar cuantificadas por las variables de $\bar{V}$ según su rango.

[4] Si $\mathcal{L}$ tiene descriptor, hemos de exigir además que exista una fórmula $\phi(x)$ sin descriptores y con x como única variable libre (de rango 0) tal que $\vdash_T \bigvee^1_0 u\, \phi(u)$, y entonces, de acuerdo con el teorema 3.29, podemos suponer que $\phi(u|u = u)$ es un axioma de T.

DEMOSTRACIÓN: Si $\mathcal{L}$ no tiene descriptor, el teorema es consecuencia inmediata de [CS 2.22], pues la extensión predicativa de T, tal y como la acabamos de definir, es la traducción de primer orden en el sentido de [CS 2.14] de la extensión predicativa de T en el sentido de [CS 2.6] (teniendo en cuenta las observaciones tras [CS 2.8]) y el teorema [CS 3.22] prueba que ésta es una extensión conservativa de T.

Si $\mathcal{L}$ tiene descriptor, el teorema 3.29 nos dice que existe una sentencia α' sin descriptores tal que $\vdash_T(\alpha \leftrightarrow \alpha')$. Entonces, como la extensión predicativa interpreta a T, en ella se demuestra $\tilde{\alpha} \leftrightarrow \tilde{\alpha}'$. Por lo tanto, si en la extensión predicativa se demuestra $\tilde{\alpha}$, también se demuestra $\tilde{\alpha}'$ y, de nuevo por 3.29, se demuestra sin descriptores, luego por el caso ya probado para lenguajes sin descriptor, en T se demuestra α', luego también α. ∎

En cualquier teoría axiomática de segundo orden T podemos distinguir entre las fórmulas estructuradas en sentido estricto (las que tenemos definidas) y las *fórmulas estructuradas en sentido amplio,* que son las equivalentes en la teoría a fórmulas estructuradas en sentido estricto. Más precisamente, una fórmula $\alpha(x_1, \ldots, x_n)$ (donde las variables libres son las indicadas y pueden ser de cualquier rango) es estructurada en sentido amplio (en T) si existe una fórmula estructurada $\beta(x_1, \ldots, x_n)$ en sentido estricto con las mismas variables libres tal que

$$\vdash_T \bigwedge\nolimits_* \bar{u}(\alpha \leftrightarrow \beta),$$

donde $\bar{u}$ liga todas las variables libres según su rango.

Así, por ejemplo, si X, Y son variables del mismo rango $r \geq 1$, la fórmula $X = Y$ no es estructurada en sentido estricto, pero por el axioma de extensionalidad (cuyo recíproco es un teorema lógico) cumple

$$\bigwedge\nolimits_r UV(U = V \leftrightarrow \bigwedge\nolimits_0 u_1 \cdots u_r(X(u_1, \ldots, u_r) \leftrightarrow Y(u_1, \ldots, u_r))),$$

y el miembro derecho es una fórmula estructurada en sentido estricto, luego $X = Y$ es estructurada en sentido amplio en cualquier teoría axiomática de segundo orden.

Otro ejemplo es la fórmula $R_r X$, que no es estructurada en sentido estricto, pero se cumple

$$\bigwedge\nolimits_r U(R_r U \leftrightarrow U = U)$$

(porque esto es lo mismo que $\bigwedge U(R_r U \to (R_r U \leftrightarrow U = U))$) y $U = U$ puede sustituirse por una fórmula estructurada en sentido estricto.

Notemos que las fórmulas estructuradas en sentido estricto no tienen descriptores, pero las estructuradas en sentido amplio sí que pueden tenerlos.

Es claro que el esquema de comprensión para fórmulas (de primer orden) estructuradas en sentido estricto implica la versión para fórmulas (de primer orden) estructuradas en sentido amplio.

Diremos que un término T es *estructurado en sentido amplio* (y de rango r) si, para cualquier variable X de rango r que no esté en T (no importa cuál), la fórmula $X = T$ es estructurada en sentido amplio. Esto incluye claramente a los términos estructurados en sentido estricto.

Notemos que, la traducción $\tilde{\alpha}$ de toda fórmula α de una teoría de primer orden T a su extensión predicativa de segundo orden $\tilde{T}$ es una fórmula estructurada en sentido estricto, pues sabemos que así es cuando la fórmula no tiene descriptores, pero si los tiene, existe una fórmula α' sin descriptores tal que

$$\underset{T}{\vdash} \bigwedge \bar{u}(\alpha \leftrightarrow \alpha'),$$

y, como $\tilde{T}$ interpreta a T, también

$$\underset{T}{\vdash} \bigwedge\nolimits_* \bar{u}(\tilde{\alpha} \leftrightarrow \tilde{\alpha}'),$$

donde $\tilde{\alpha}'$ es estructurada en sentido estricto.

Similarmente, las traducciones de todos los términos de T son términos estructurados en sentido estricto.

Conjuntos En la extensión predicativa de una teoría axiomática de primer orden podemos identificar las relaciones monádicas con conjuntos. Podemos escribir

$$\mathrm{cto}\, T \equiv R_1 T, \qquad t \in T \equiv T(t), \qquad t \notin T \equiv \neg T(t),$$

y llamar

$$\{u \mid \phi(u)\} \equiv U|(\mathrm{cto}\, U \wedge \bigwedge\nolimits_0 u(u \in U \leftrightarrow \phi(u))).$$

Así, si $\phi(x, \bar{X})$ es una fórmula estructurada de primer orden (o de segundo orden, si consideramos la extensión plena en lugar de la predicativa), el axioma de extensionalidad nos asegura que el conjunto cuya existencia asegura el axioma de comprensión es único, por lo que la descripción anterior es propia y tenemos que

$$\bigwedge\nolimits_* \bar{U}(\mathrm{cto}\{u \mid \phi(u, \bar{U})\} \wedge \bigwedge\nolimits_0 u(u \in \{u \mid \phi(u, \bar{U})\} \leftrightarrow \phi(u, \bar{U}))).$$

En particular, podemos definir

1. $X \cup Y \equiv \{u \mid u \in X \vee u \in Y\}$,
2. $X \cap Y \equiv \{u \mid u \in X \wedge u \in Y\}$,
3. $X \setminus Y \equiv \{u \mid u \in X \wedge u \notin Y\}$,
4. $\bar{X} \equiv \{u \mid u \notin X\}$,
5. $V \equiv \{u|u = u\}$,
6. $\varnothing \equiv \{u|u \neq u\}$,
7. $\{x_1, \ldots, x_n\} \equiv \{u \mid u = x_1 \vee \cdots \vee u = x_n\}$,

de modo que todas las descripciones son propias cuando X e Y son conjuntos y $x_1, \ldots, x_n$ son objetos (es decir, cumplen $R_0 x_i$), y se cumplen las propiedades obvias. Además, todos los términos son estructurados.

También podemos definir la inclusión:

$$X \subset Y \equiv \bigwedge_0 u(u \in X \to u \in Y),$$

y demostrar las propiedades obvias, como que

$$\bigwedge_1 UV(U \subset V \wedge V \subset U \to U = V),$$

que es una forma equivalente del axioma de extensionalidad.

Sin embargo, no podemos definir un par ordenado como $(x, y) = \{\{x\}, \{x, y\}\}$, porque un conjunto $\{x\}$ no pertenece a otro conjunto. Pero las relaciones diádicas hacen que no necesitemos pares ordenados.

Funciones Definimos el *producto cartesiano* de n conjuntos como

$$X_1 \times \cdots \times X_n \equiv U|(R_n U \wedge \bigwedge u_1 \cdots u_n(U(u_1, \ldots, u_n) \leftrightarrow u_1 \in X_1 \wedge \cdots \wedge u_n \in X_n)).$$

Es claro que si $X_1, \ldots, X_n$ son conjuntos, entonces la descripción es propia, pues el axioma de comprensión implica la existencia de U y el de extensionalidad la unicidad.

Definimos el *dominio* y el *rango* de una relación $r + 1$-ádica R (con $r \geq 1$) como

$$\mathcal{D}_r R \equiv U|(R_r U \wedge \bigwedge_0 u_1 \cdots u_r(U(u_1, \ldots, u_r) \leftrightarrow \bigvee_0 v\, R(u_1, \ldots, u_r, v))),$$

$$\mathcal{R}_r R \equiv U|(R_1 U \wedge \bigwedge_0 v(U(v) \leftrightarrow \bigvee_0 u_1 \cdots u_r\, R(u_1, \ldots, u_r, v))).$$

Ambas descripciones son propias (cuando $R_{r+1}R$), de nuevo por los axiomas de comprensión y extensionalidad.

Definimos

$$F : X_1 \times \cdots \times X_n \longrightarrow Y \equiv R_{n+1}F \wedge \mathcal{D}_r F = X_1 \times \cdots \times X_n \wedge \mathcal{R}_r F \subset Y \wedge$$
$$\bigwedge_0 u_1 \cdots u_n(u_1 \in X_1 \wedge \cdots \wedge u_n \in X_n \to \bigvee_0 v(v \in Y \wedge F(u_1, \ldots, u_n, v)) \wedge$$
$$\bigwedge_0 u_1 \cdots u_n v v'(F(u_1, \ldots, u_n, v) \wedge F(u_1, \ldots, u_n, v') \to v = v')).$$

Así como $F(x_1, \ldots, x_n) \equiv u|(R_0 u \wedge F(x_1, \ldots, x_n, u))$, que es un término estructurado, ya que

$$\bigwedge_{n+1} F \bigwedge_0 u_1 \cdots u_n v(v = F(u_1, \ldots, u_n) \leftrightarrow \bigwedge_0 v'(F(u_1, \ldots, u_n, v') \leftrightarrow v' = v)$$
$$\vee (\neg \bigvee_0 w(\bigwedge_0 v'(F(u_1, \ldots, u_n, v') \leftrightarrow v' = w)) \wedge \tilde{\phi}(v))),$$

donde $\phi(x)$ es la fórmula sin descriptores que caracteriza la descripción impropia, y $\tilde{\phi}$ es su traducción (que es una fórmula estructurada).

De este modo podemos hablar de funciones, podemos definir la composición de funciones, etc., pero en lugar de trabajar en este contexto general vamos a considerar en la sección siguiente el caso de las teorías aritméticas que conocemos.

En resumen, lo que hemos probado en esta sección es que a cualquier teoría axiomática de primer orden le podemos añadir variables de segundo orden para convertirla en una teoría de segundo orden en la que las relaciones satisfacen el axioma de extensionalidad y el axioma de comprensión para fórmulas estructuradas de primer orden. La teoría así obtenida es una extensión conservativa de la teoría de partida En cambio, puede probarse que la extensión plena de segundo orden ya no es, en general, una extensión conservativa de una teoría de primer orden dada.

7.2 La aritmética de segundo orden

Como en la aritmética de Peano podemos definir n-tuplas, es más práctico considerar extensiones de primer orden reducidas, con lo que sólo tenemos dos clases de objetos: los números naturales (los objetos de primer orden) y los conjuntos (las relaciones monádicas u objetos de segundo orden). Aunque para la definición siguiente podríamos basarnos en las definiciones de la sección precedente, vamos a darla de forma totalmente explícita por claridad:

Definición 7.5 El *lenguaje de la aritmética de segundo orden* es el lenguaje formal de segundo orden reducido $\mathcal{L}_a^2$ cuyos signos (además de las variables, los conectores, los cuantificadores y el descriptor) son

1. La constante 0.
2. El funtor monádico S y los funtores diádicos $+$ y $\cdot$.
3. El igualador $=$, el relator diádico $\leq$ y los relatores estructurales R_1 (monádico) y S_1 (diádico).

Adoptaremos los convenios de notación siguientes:

$$t' \equiv St, \quad \mathrm{cto}\, t \equiv R_1 t, \quad \mathrm{Nat}\, t \equiv \neg\, \mathrm{cto}\, t, \quad t \in T \equiv S_1(T,t), \quad t \notin T \equiv \neg t \in T.$$

Además, usaremos letras minúsculas para las variables de primer orden y letras mayúsculas para las de segundo orden, con lo que la notación para los cuantificadores será:

$$\bigwedge u\, \alpha \equiv \bigwedge\nolimits_0 u\, \alpha \equiv \bigwedge u(\mathrm{Nat}\, u \to \alpha), \quad \bigvee u\, \alpha \equiv \bigvee\nolimits_0 u\, \alpha \equiv \bigvee u(\mathrm{Nat}\, u \wedge \alpha),$$

$$\bigwedge U\, \alpha \equiv \bigwedge\nolimits_1 U\, \alpha \equiv \bigwedge U(\mathrm{cto}\, U \to \alpha), \quad \bigvee U\, \alpha \equiv \bigvee\nolimits_1 U\, \alpha \equiv \bigvee U(\mathrm{cto}\, U \wedge \alpha).$$

Los *semitérminos de primer orden* se definen por las reglas siguientes:

1. Las variables de primer orden son semitérminos de primer orden.
2. 0 es un semitérmino de primer orden.
3. Si t_1, t_2 son semitérminos de primer orden, también lo son t_1', $t_1 + t_2$, $t_1 \cdot t_2$.

Las *semifórmulas estructuradas* se definen por las reglas siguientes:

1. Si t_1, t_2 son semitérminos de primer orden y X es una variable de segundo orden (libre o ligada), las semifórmulas $t_1 = t_2$, $t_1 \leq t_2$, $t_1 \in X$ son semifórmulas estructuradas.

2. Si α y β son semifórmulas estructuradas, también lo son

$$\neg\alpha, \quad \alpha \vee \beta, \quad \bigwedge u\,\alpha, \quad \bigvee u\,\alpha, \quad \bigwedge U\,\alpha, \quad \bigvee U\,\alpha$$

Las semifórmulas estructuradas sin variables ligadas de segundo orden se llaman *fórmulas aritméticas.*

La *aritmética de Peano de segundo orden*AP^2 es la teoría axiomática de segundo orden cuyos axiomas son los indicados en la página siguiente.

Si limitamos el esquema de comprensión a fórmulas aritméticas tenemos la teoría llamada ACA_0 (por "axioma de comprensión aritmética"). Vamos a ver que AP^2 y ACA_0 son las extensiones plena y predicativa, respectivamente, de AP, salvo por el principio de inducción, que es más fuerte que el que tendríamos en estas teorías.

En efecto, recordemos ante todo la definición de traducción de una fórmula de $\mathcal{L}_a$ a $\mathcal{L}_a^2$, que no es sino la fórmula que resulta de reemplazar cada cuantificador $\bigwedge u$ o $\bigvee u$ por $\bigwedge u(\mathrm{Nat}\,u \to \ldots)$ o $\bigvee u(\mathrm{Nat}\,u \wedge \cdots)$, respectivamente. Con el convenio que hemos introducido de usar letras minúsculas para los cuantificadores restringidos a números naturales, la traducción de una fórmula de $\mathcal{L}_a$ se expresa sin cambio alguno siempre y cuando escribamos sus variables con letras minúsculas.

Es claro entonces que las traducciones de fórmulas de $\mathcal{L}_a$ sin descriptores son precisamente las fórmulas estructuradas sin variables de segundo orden.

Ahora, para cada fórmula estructurada $\alpha(x, \bar{y}, \bar{Y})$, llamamos

$$\mathrm{Ind}(\alpha) \equiv \bigwedge \bar{V} \bigwedge \bar{v}(\alpha(0, \bar{v}, \bar{V}) \wedge \bigwedge u(\alpha(u, \bar{v}, \bar{V}) \to \alpha(u', \bar{v}, \bar{V})) \to \bigwedge u\,\alpha(u, \bar{v}, \bar{V})),$$

de modo que la traducción del principio de inducción para una fórmula α de $\mathcal{L}_a$ es la sentencia $\mathrm{Ind}(\tilde{\alpha})$, donde $\tilde{\alpha}$ es la traducción de α. Vemos entonces que, tanto la extensión predicativa de AP como la extensión plena, tienen el esquema de inducción formado por las sentencias $\mathrm{Ind}(\alpha)$ para todas las fórmulas estructuradas sin variables de segundo orden.

En cambio, el principio de inducción de la definición precedente es equivalente en ACA_0 (resp. en AP^2) al esquema formado por las sentencias $\mathrm{Ind}(\alpha)$ para todas las fórmulas aritméticas (resp. para todas las fórmulas estructuradas).[5]

En efecto, por una parte, el principio de inducción es el caso particular del esquema correspondiente a la fórmula aritmética $\alpha(x, X) \equiv X(x)$, mientras que, a partir del principio de inducción que acabamos de introducir, podemos probar todas las sentencias $\mathrm{Ind}(\alpha)$ como sigue:

[5]Esto es lo que expresa el subíndice 0 en ACA_0, que la inducción está restringida a fórmulas aritméticas. La teoría ACA es la que resulta de añadir al ACA_0 el esquema $\mathrm{Ind}(\alpha)$ para todas las fórmulas estructuradas.

Axiomas de la aritmética de Peano de segundo orden

Extensionalidad $\bigwedge UV(\bigwedge u(u \in U \leftrightarrow u \in V) \to U = V)$

Comprensión $\bigwedge \bar{V} \bigwedge \bar{v} \bigvee U \bigwedge u(u \in U \leftrightarrow \phi(u, \bar{v}, \bar{V}))$,

para toda fórmula estructurada $\phi(x, \bar{y}, \bar{Y})$ sin más variables libres que las indicadas.

AP1 $\mathrm{Nat}\, 0$

AP2 $\bigwedge u\, \mathrm{Nat}\, u'$

AP3 $\bigwedge u\, u' \neq 0$

AP4 $\bigwedge uv(u' = v' \to u = v)$

Inducción $\bigwedge U(0 \in U \wedge \bigwedge u(u \in U \to u' \in U) \to \bigwedge u\, u \in U)$

S1 $\bigwedge u\, u + 0 = u$

S2 $\bigwedge uv\ (u + v') = (u + v)'$

M1 $\bigwedge u\, u \cdot 0 = 0$

M2 $\bigwedge uv\ u \cdot v' = u \cdot v + u$

O1 $\bigwedge u\, u \leq u$

O2 $\bigwedge uv(u \leq v \wedge v \leq u \to u = v)$

O3 $\bigwedge uvw(u \leq v \wedge v \leq w \to u \leq w)$

O4 $\bigwedge uv(u \leq v \vee v \leq u)$

O5 $\bigwedge u\, u \leq u'$

Incluimos además el axioma "semilógico" $0 = u | u = u$.

Por el principio de comprensión (y aquí tenemos que suponer que α es aritmética en el caso de ACA_0) existe un conjunto X tal que $\bigwedge u(u \in X \leftrightarrow \alpha(u, \bar{y}, \bar{Y}))$ y, aplicando el principio de inducción a este conjunto X, obtenemos $\mathrm{Ind}(\alpha)$.

La diferencia es sustancial. De poco serviría disponer de conjuntos si tuviéramos prohibido aplicar el principio de inducción a fórmulas en las que aparezcan conjuntos.

Por ejemplo, usando los principios de inducción y comprensión podemos probar (en ACA_0)

$$\bigwedge uv\, \mathrm{Nat}(u + v), \qquad \bigwedge uv\, \mathrm{Nat}(u \cdot v).$$

Para ello observamos que $\mathrm{Nat}(x + y)$ no es, según la definición, una fórmula estructurada, pero es equivalente $\alpha(y) \equiv \bigvee v(v = x + y)$, que sí que lo es, lo que

hace que

$$\bigwedge v(\mathrm{Nat}(v+0) \wedge \bigwedge u(\mathrm{Nat}(v+u) \to \mathrm{Nat}(v+u')) \to \bigwedge u\mathrm{Nat}(v+u))$$

sea equivalente a $\mathrm{Int}(\alpha)$, luego es un teorema de ACA_0, que legitima la inducción trivial que prueba que la suma de números naturales es un número natural. Usando este hecho se prueba el correspondiente al producto.

Usando estos dos teoremas junto con AP1 y AP2, una simple inducción sobre la longitud de un término de primer orden $t(\bar{x}, \bar{X})$, prueba que $\bigwedge \bar{U} \bigwedge \bar{u}\ \mathrm{Nat}\, t$, que es el primer esquema estructural de la definición 7.3, del cual AP1 y AP2 son casos particulares. El segundo lo hemos incluido entre los axiomas de ACA_0 y el tercero es trivial en las extensiones reducidas por la definición de R_1 como la negación de R_2.

Así pues, lo único que tienen de más ACA_0 o AP^2 respecto a las extensiones predicativa y plena de AP es la forma general del principio de inducción. Esto hace que no podamos aplicar el teorema 7.4 para concluir que ACA_0 es una extensión conservativa de AP, pero aun así, el resultado es cierto:

Teorema 7.6 *La teoría de segundo orden* ACA_0 *es una extensión conservativa de* AP, *es decir, una fórmula es demostrable en* AP *si y sólo si su traducción a* $\mathcal{L}_a^2$ *es demostrable en* ACA_0.

Demostración: Basta tener en cuenta que ACA_0 es la traducción de primer orden (en el sentido de [CS 2.14]) del cálculo secuencial de segundo orden ACA_0 definido en [CS 2.16], que es una extensión conservativa de AP por [CS 3.23]. Esto prueba que, para fórmulas α sin descriptores, la traducción de α es demostrable en ACA_0 si y sólo si α es demostrable en AP. Si α tiene descriptores razonamos como en la prueba del teorema 7.4. ∎

Sin embargo, en el contexto de este libro estamos más interesados en obtener una extensión conservativa de segundo orden de $\mathrm{I}\Sigma_1$ (y, por consiguiente, de la aritmética recursiva primitiva). Como es fundamental conservar la posibilidad de usar conjuntos en los razonamientos inductivos, necesitamos generalizar como sigue la jerarquía de Kleene:

Definición 7.7 Las semifórmulas de tipo Δ_0^0 son las fórmulas de $\mathcal{L}_a^2$ definidas por las reglas siguientes [CS 2.3]:

1. Si t_1, t_2 son semitérminos de primer orden, entonces $t_1 = t_2$ y $t_1 \le t_2$ son semifórmulas Δ_0^0.
2. Si t es un semitérmino de primer orden y X es una variable de segundo orden (libre o ligada), entonces $X(t)$ es una semifórmula Δ_0^0.
3. Si α y β son semifórmulas Δ_0^0, también lo son $\neg\alpha$ y $\alpha \vee \beta$.
4. Si t es un semitérmino de primer orden y α es una semifórmula Δ_0^0, también lo son $\bigwedge u \le t\, \alpha$ y $\bigvee u \le t\, \alpha$.

Las fórmulas Σ^0_n y Π^0_n se definen igual que las fórmulas Σ_n y Π_n (definición 4.20), pero partiendo de fórmulas Δ^0_0 en lugar de Δ_0.

De este modo, las fórmulas Σ_n y Π_n son las fórmulas Σ^0_n y Π^0_n que no tienen variables de segundo orden, mientras que las fórmulas Σ^0_n y Π^0_n pueden tener variables libres (pero no ligadas) de segundo orden.

En estos términos, nos interesaría una extensión conservativa de $\mathrm{I}\Sigma_1$ que admitiera la inducción sobre fórmulas de tipo Σ^0_1. Podría pensarse en que para ello basta cambiar el principio de inducción de ACA_0 por el esquema $\mathrm{Ind}(\alpha)$, para fórmulas de tipo Σ^0_1, pero resulta que eso no funciona —como pronto veremos— sino que es necesario restringir también el axioma de comprensión.

Definición 7.8 Llamaremos[6] ACR_0 a la teoría axiomática de segundo orden sobre $\mathcal{L}^2_a$ cuyos axiomas son los mismos de ACA_0 excepto los axiomas de comprensión y de inducción, que se sustituyen, respectivamente, por los esquemas

$\boldsymbol{\Delta^0_1}$-comprensión $\bigwedge \bar{V}\bar{v}(\bigwedge u(\alpha(u) \leftrightarrow \beta(u)) \to \bigvee U \bigwedge u(u \in U \leftrightarrow \alpha(u)))$,
para todas las fórmulas $\alpha(x, \bar{y}, \bar{Y})$ de tipo Σ^0_1 y $\beta(x, \bar{y}, \bar{Y})$ de tipo Π^0_1.

$\boldsymbol{\Sigma^0_1}$-inducción $\bigwedge \bar{V} \bigwedge \bar{v}(\alpha(0) \wedge \bigwedge u(\alpha(u) \to \alpha(u')) \to \bigwedge u\, \alpha(u))$,
para toda fórmula $\alpha(x, \bar{y}, \bar{Y})$ de tipo Σ^0_1.

Así pues, en ACR_0, para que una fórmula defina un conjunto, tiene que ser de tipo Σ^0_1 y además equivalente a otra de tipo Π^0_1, es decir, tiene que ser de tipo Δ^0_1, aunque esto es relativo a la teoría en la cual puede probarse la equivalencia, de modo que en el axioma simplemente hemos puesto la equivalencia como hipótesis.

La prueba que hemos dado de que en ACA_0 se pueden demostrar los axiomas estructurales de la lógica de segundo orden vale igualmente en ACR_0, por lo que esta teoría extiende, de hecho, a la extensión predicativa de $\mathrm{I}\Sigma_1$. En particular, en ella podemos identificar cada número natural con una sucesión finita, lo cual hace a su vez que en ella podamos probar (con la misma prueba) el teorema 4.21 para fórmulas Σ^0_n y Π^0_n, así como 4.27, es decir, que en ACR_0 se puede demostrar que si α es una fórmula de tipo Σ^0_1 (resp. Π^0_1) y t es un término de primer orden, entonces $\bigwedge u \le t\, \alpha$ y $\bigvee u \le t\, \alpha$ son equivalentes a fórmulas de tipo Σ^0_1 (resp. Π^0_1) y si añadimos a ACR_0 el principio de inducción para fórmulas Σ^0_n, también podemos probar que las fórmulas de tipo Σ^0_n y Π^0_n son cerradas para cuantificadores acotados en este sentido.

El teorema siguiente explica por qué es necesario restringir el principio de comprensión:

Teorema 7.9 *Si a* ACR_0 *le añadimos el principio de* Σ^0_1*-comprensión, en la teoría resultante podemos demostrar todos los teoremas de* ACA_0.

[6]Las siglas corresponden a "axioma de comprensión recursiva" y el subíndice indica que la inducción está restringida a fórmulas de tipo Σ^0_1. ACR es la teoría que resulta de extender el esquema de inducción a todas las fórmulas estructuradas.

DEMOSTRACIÓN: Veamos en primer lugar que en ACR_0 más Σ^0_1-comprensión podemos demostrar el axioma de comprensión completo. Para ello veamos en primer lugar que el axioma de Σ^0_n-comprensión implica el de Π^0_n-comprensión.

Si $\alpha(x, \bar{y}, \bar{Y})$ es una fórmula de tipo Π^0_n, es claro que $\neg\alpha$ es equivalente a una fórmula de tipo Σ^0_n, luego, por Σ^0_n-comprensión, fijados números $\bar{y}$ y conjuntos $\bar{Y}$, tenemos que existe un conjunto X tal que

$$\bigwedge u(u \in X \leftrightarrow \neg\alpha(u, \bar{y}, \bar{X})).$$

Ahora, por Δ^0_0-comprensión (que es más débil que la Δ^0_1-comprensión), existe un conjunto Y tal que $\bigwedge u(u \in Y \leftrightarrow u \notin X)$, luego $\bigwedge u(u \in Y \leftrightarrow \alpha(u, \bar{y}, \bar{X}))$, lo que prueba la Π^0_n comprensión.

Ahora veamos que la Σ^0_n-comprensión implica la Σ^0_{n+1}-comprensión. Para ello tomamos una fórmula $\alpha(x, \bar{y}, \bar{Y}) \equiv \bigvee u\, \beta(u, x, \bar{y}, \bar{Y})$, de tipo Σ^0_{n+1}, con lo que β es de tipo Π^0_n.

Como ya hemos probado que la Σ^0_n-comprensión implica la Π^0_n-comprensión, tenemos que, fijados $\bar{y}$, $\bar{Y}$, existe un conjunto X tal que

$$\bigwedge w(w \in X \leftrightarrow \bigwedge st \le w(w = \langle s, t\rangle \to \beta(s, t, \bar{y}, \bar{Y}))),$$

pues la fórmula de la izquierda es ciertamente Π^0_n (ya que $w = \langle s, t\rangle$ es Δ_1, luego también Π^0_1). En particular:

$$\bigwedge st(\langle s, t\rangle \in X \leftrightarrow \beta(s, t, \bar{y}, \bar{Y})).$$

A su vez, por Σ^0_1-especificación existe un conjunto Y tal que

$$\bigwedge(w \in Y \leftrightarrow \bigvee up(p = \langle u, w\rangle \wedge p \in X)),$$

que equivale a

$$\bigwedge w(w \in Y \leftrightarrow \bigvee u\, \beta(u, w, \bar{y}, \bar{Y})),$$

lo que prueba la Σ_{n+1}-comprensión.

Así pues, en $\mathrm{ACR}_0 + \Sigma^0_1$-especificación podemos probar la Σ^0_n-especificación para todo n, luego podemos probar todos los casos del esquema de especificación de ACA_0. Por otro lado, el principio de Σ^0_1-inducción aplicado a la fórmula $\alpha(x, X) \equiv X(x)$ (de tipo Δ^0_0) no es sino el principio de inducción de ACA_0. ∎

Así pues, para aspirar a tener una extensión conservativa de $\mathrm{I}\Sigma_n$, necesitamos restringir el axioma de comprensión a fórmulas Δ^0_1. Y esto es suficiente:

Teorema 7.10 *La teoría* ACR_0 *más el principio de* Σ^0_n*-inducción es una extensión conservativa de* $\mathrm{I}\Sigma_n$*, es decir, que una fórmula de* $\mathcal{L}_a$ *es demostrable en* $\mathrm{I}\Sigma_n$ *si y sólo si su traducción a* $\mathcal{L}^2_a$ *es demostrable en la teoría indicada.*

DEMOSTRACIÓN: Basta tener en cuenta que ACR_0 es la traducción de primer orden en el sentido de [CS 2.14] de la teoría considerada en [CS 2.18], y el teorema [CS 2.34] prueba que es una extensión conservativa de $\mathrm{I}\Sigma_n$. Para tratar las fórmulas con descriptores basta considerar el argumento empleado en la prueba de 7.4. ∎

Ahora observamos que el mismo argumento empleado en la prueba del teorema 4.24 implica el teorema siguiente:

Teorema 7.11 *En* ACR_0*, el principio de* Σ^0_n*-inducción implica el principio de* Π^0_n*-inducción.*

Similarmente, la prueba del teorema 4.29 se adapta trivialmente para probar:

Teorema 7.12 *Si* $\phi(x)$ *es una fórmula de tipo* Σ^0_n *o* Π^0_n*, en* ACR_0 *más el principio de* Σ^0_n*-inducción se prueba que*

$$\bigwedge \bar{V} \bigwedge \bar{v} (\bigvee u \phi(u) \to \overset{1}{\bigvee} u (\phi(u) \wedge \bigwedge w < u \, \neg \phi(w))).$$

Terminamos esta sección con un resultado que necesitaremos para la prueba del teorema [CS 2.34], que hemos usado en la demostración de 7.10:

Definición 7.13 Llamamos ARP^2_0 a la teoría axiomática de segundo orden sobre $\mathcal{L}^2_{\mathrm{arp}}$ cuyos axiomas son:

Extensionalidad $\bigwedge UV (\bigwedge u (u \in U \leftrightarrow u \in V) \to U = V)$

Δ^0_0-Comprensión $\bigwedge \bar{V} \bigwedge \bar{v} \bigvee U \bigwedge u (u \in U \leftrightarrow \phi(u, \bar{v}, \bar{V}))$,

para toda fórmula estructurada $\phi(x, \bar{y}, \bar{Y})$ de tipo Δ^0_0 sin más variables libres que las indicadas.[7]

AP1 $\mathrm{Nat}\, 0$

AP2 $\bigwedge u \, \mathrm{Nat}\, Su$

AP3 $\bigwedge u \, Su \neq 0$

AP4 $\bigwedge uv (Su = Sv \to u = v)$

Σ^0_1-inducción $\bigwedge \bar{V} \bar{v} (\gamma(0) \wedge \bigwedge u (\gamma(u) \to \gamma(Su)) \to \bigwedge u \, \gamma(u))$,

para toda fórmula $\gamma(x, \bar{y}, \bar{Y})$ de tipo Σ^0_1.

más las definiciones de los funtores de $\mathcal{L}_{\mathrm{arp}}$ distintos de S (definición 1.8).

Como en el caso de ACA_0, a partir de estos axiomas podemos demostrar los axiomas estructurales de la definición 7.2, por lo que ARP^2_0 es una teoría axiomática de segundo orden según dicha definición.

Teorema 7.14 *La teoría* ARP^2_0 *es una extensión intrascendente de la teoría* ACR^-_0 *que resulta de restringir en* ACR_0 *el axioma de comprensión a fórmulas de tipo* Δ^0_0*.*

[7] Aquí hay que entender que las fórmulas Δ^0_0 son las dadas por la definición análoga a 7.7 para el lenguaje $\mathcal{L}^2_{\mathrm{arp}}$, es decir, las que tienen todos sus cuantificadores acotados por términos, en este caso de $\mathcal{L}^2_{\mathrm{arp}}$, que son los mismos que los términos de $\mathcal{L}_{\mathrm{arp}}$.

DEMOSTRACIÓN: La prueba es esencialmente la misma que la del teorema 4.49, así que nos limitaremos a esbozar los cambios. Allí hemos visto que a cada término t de $\mathcal{L}_{\mathrm{arp}}$ (y los términos de $\mathcal{L}_{\mathrm{arp}}$ son los mismos que los de $\mathcal{L}^2_{\mathrm{arp}}$), le podemos asignar una fórmula $\psi_t(y)$ de $\mathcal{L}_a$, de tipo $\Delta_1^{\mathrm{I}\Sigma_1}$ de modo que

$$\underset{\mathrm{I}\Sigma_1}{\vdash} \overset{1}{\bigvee} y\, \psi_t(y), \qquad \underset{\mathrm{I}\Sigma_1^+}{\vdash} (y = t \leftrightarrow \psi_t(y)).$$

A continuación definíamos una traducción α^* de cada semifórmula de $\mathcal{L}_{\mathrm{arp}}$, y ahora sólo tenemos que extender la definición a semifórmulas de $\mathcal{L}^2_{\mathrm{arp}}$. Para ello basta definir

$$(\mathrm{cto}\, t)^* \equiv \bigvee u(\psi_t(u) \wedge \mathrm{cto}\, u), \quad (t \in X)^* \equiv \bigvee u(\psi_t(u) \wedge u \in X).$$

Así, para cada fórmula α de $\mathcal{L}^2_{\mathrm{arp}}$, tenemos que α^* es una fórmula de $\mathcal{L}^2_a$ con las mismas variables libres (y en la misma posición en la jerarquía de Kleene) y tal que $\underset{\mathrm{ARP}^2}{\vdash}(\alpha \leftrightarrow \alpha^*)$. Si α está en $\mathcal{L}^2_a$ la equivalencia se demuestra en ACR_0^-.

Luego se prueba que las traducciones de los axiomas de ARP_0^2 son teoremas de ACR_0^-, de donde se sigue que

$$\text{si } \underset{\mathrm{ARP}_0^2}{\vdash} \alpha, \quad \text{entonces} \quad \underset{\mathrm{ACR}_0^-}{\vdash} \alpha^*,$$

con lo que si α es una fórmula (sin descriptores) de $\mathcal{L}^2_a$ y es demostrable en ARP_0^2, también lo es en ACR_0^-. ∎

7.3 El axioma de comprensión recursiva

Vamos a probar algunos hechos básicos en ACR_0. Algunos conceptos los hemos introducido ya en la sección 7.1 en el contexto general de la lógica predicativa, pero los reintroducimos aquí por claridad. Es el caso de la inclusión:

$$X \subset Y \equiv \bigwedge u(u \in X \to u \in Y),$$

que permite probar las propiedades obvias:

$$\bigwedge X\, X \subset X, \qquad \bigwedge XY(X \subset Y \wedge Y \subset X \to X = Y),$$
$$\bigwedge XYZ(X \subset Y \wedge Y \subset Z \to X \subset Z).$$

Usaremos la notación:

$$\{u \mid \alpha(u)\} \equiv U \mid \bigwedge u(u \in U \leftrightarrow \alpha(u)).$$

El axioma de Δ_1^0-comprensión, junto con el axioma de extensionalidad, implican que, si α y β son fórmulas de tipo Σ_1^0 y Δ_1^0, entonces

$$\bigwedge \bar{V}\bar{v}(\bigwedge u(\alpha(u) \leftrightarrow \beta(u)) \to \overset{1}{\bigvee} U \bigwedge u(u \in U \leftrightarrow \alpha(u))),$$

de donde se sigue a su vez que

$$\bigwedge \bar{V}\bar{v}(\bigwedge u(\alpha(u) \leftrightarrow \beta(u)) \to \bigwedge u(u \in \{u \mid \alpha(u)\} \leftrightarrow \alpha(u))).$$

Esto es aplicable claramente a los casos siguientes, en los que las fórmulas consideradas son de tipo Δ_0^0:

$$X \cup Y \equiv \{u \mid u \in X \vee u \in Y\}, \qquad X \cap Y \equiv \{u \mid u \in X \wedge u \in Y\},$$

$$\bar{X} \equiv \{u \mid u \notin X\}, \qquad X \setminus Y \equiv \{u \mid u \in X \wedge u \notin Y\},$$

$$\mathbb{N} \equiv \{u \mid u = u\}, \qquad \varnothing \equiv \{u \mid u \neq u\}, \qquad I_n \equiv \{u \mid u < n\},$$

$$\{x_1, \ldots, x_n\} \equiv \{u \mid u = x_1 \vee \cdots \vee u = x_n\}.$$

A partir de estas definiciones podemos probar los resultados obvios, como

$$\bigwedge XYZ\,(X \cup (Y \cap Z) = (X \cup Y) \cap (X \cup Z)),$$

etc. Recordemos que tenemos definida (definición 4.17) la fórmula $z = \langle x, y\rangle_2$, que es equivalente a

$$2z = (x + y)(x + y + 1) + 2x,$$

y que en $\mathrm{I}\Sigma_1$, luego en ACR_0, se prueba que esta relación hace corresponder cada número natural z con un par ordenado de números naturales (z_1, z_2), de modo que $z_1, z_2 \leq z$.

Ahora podemos definir el producto cartesiano:

$$X \times Y \equiv \{u \mid \bigvee vw \leq u(v \in X \wedge w \in Y \wedge u = \langle v, w\rangle_2)\}.$$

La fórmula $u = \langle v, w\rangle_2$ es Δ_0, lo que nos permite aplicar el principio de Δ_1^0-comprensión. Así

$$\bigwedge XY \bigwedge u(u \in X \times Y \leftrightarrow u_1 \in X \wedge u_2 \in Y).$$

Funciones A su vez, ahora podemos formalizar el concepto general de función:

$$F : X \longrightarrow Y \equiv F \subset X \times Y \wedge \bigwedge u \bigvee v(u \in X \to \langle u, v\rangle_2 \in F) \wedge$$

$$\bigwedge uvw(\langle u, v\rangle_2 \in F \wedge \langle u, w\rangle_2 \in F \to v = w).$$

Así tenemos que $F : X \longrightarrow Y \wedge x \in X \to \overset{1}{\bigvee} v(v \in Y \wedge \langle x, u\rangle_2 \in F)$. Por lo tanto, si definimos

$$F(x) \equiv v \mid \langle x, v\rangle_2 \in F,$$

tenemos que

$$\bigwedge FXY(F : X \longrightarrow Y \wedge x \in X \to F(x) \in Y \wedge \langle x, F(x)\rangle_2 \in F).$$

También es fácil ver que

$$\bigwedge FXY(F : X \longrightarrow Y \wedge G : X \longrightarrow Z \wedge \bigwedge u(u \in X \to F(u) = G(u)) \to F = G).$$

Definimos la restricción $F|_A \equiv F \cap (A \times \mathbb{N})$, de modo que

$$\bigwedge FXYA(F : X \longrightarrow Y \wedge A \subset X \to F|_A : A \longrightarrow Y).$$

Definimos también la composición de funciones como

$$F \circ G \equiv \{p \mid \bigvee v \bigvee uw < p(\langle u, v\rangle_2 \in F \wedge \langle v, w\rangle_2 \in G \wedge p = \langle u, w\rangle_2)\}.$$

Para que esta definición funcione correctamente hay que justificar que la fórmula que define $F \circ G$ es Δ^0_1. En efecto:

Teorema 7.15 $\bigwedge FGXYZ(F : X \longrightarrow Y \wedge G : Y \longrightarrow Z \rightarrow$
$F \circ G : X \longrightarrow Z \wedge \bigwedge u \in X\, (F \circ G)(u) = G(F(u))).$

Demostración: En principio, la fórmula que define a $F \circ G$ es Σ^0_1, pero, bajo las hipótesis del teorema, es equivalente a

$$\bigvee uw < p(p = \langle u, w\rangle_2 \wedge u \in X \wedge \bigwedge v(\langle u, v\rangle_2 \in F \rightarrow \langle v, w\rangle_2 \in G)),$$

que es Π^0_1, por lo que podemos aplicar el principio de Δ^0_1-especificación, y a partir de ahí se obtiene la conclusión sin dificultad. ∎

Definimos:

$F : X \longrightarrow Y$ inyectiva $\equiv F : X \longrightarrow Y \wedge \bigwedge uv \in X(F(u) = F(v) \rightarrow u = v)$,

$F : X \longrightarrow Y$ suprayecgtiva $\equiv F : X \longrightarrow Y \wedge \bigwedge v \in Y \bigvee u \in X\, F(u) = v$.

$F : X \longrightarrow Y$ biyectiva $\equiv F : X \longrightarrow Y$ inyectiva y suprayectiva.

Además:

$$X^{-1} \equiv \{u \mid \bigvee vw \leq u\, u = \langle v, w\rangle_2 \wedge \langle w, v\rangle_2 \in X\}.$$

Con lo que podemos probar hechos obvios, como que si $F : X \longrightarrow Y$ biyectiva, entonces $F^{-1} : Y \longrightarrow X$ biyectiva, etc.

Definimos la identidad $I \equiv \{u \mid \bigvee v \leq u\, u = \langle v, v\rangle_2\}$, de modo que

$$I|_A : A \longrightarrow A \text{ biyectiva.}$$

Veamos ahora que podemos definir funciones de forma recurrente:

Teorema 7.16 (Recursión) *Si $G : X \longrightarrow Y$ y $H : X \times \mathbb{N} \times Y \longrightarrow Y$, existe una única función $F : X \times \mathbb{N} \longrightarrow Y$ tal que*

$$\bigwedge u \in X\ F(u, 0) = G(u), \qquad \bigwedge u \in X \bigwedge n \in \mathbb{N}\ F(u, n+1) = H(u, n, F(n)).$$

Demostración: Basta definir

$$F \equiv \{u \mid \bigvee s \bigvee vnw \leq u(u = \langle\langle v, n\rangle_2, w\rangle_2 \wedge v \in X \wedge \ell(s) = n+1 \wedge$$

$$s_0 = G(v) \wedge \bigwedge i < n\ s_{i+1} = H(v, i, s_i) \wedge w = s_n)\}$$

y observar que la fórmula que define a F es[8] Δ^0_1, pues en principio es Σ^0_1, pero equivale a

$$\bigvee vnw \le u(u = \langle\langle v, n\rangle_2, w\rangle_2 \wedge v \in X \wedge \bigwedge s(\ell(s) = n+1 \wedge$$
$$s_0 = G(v) \wedge \bigwedge i < n\ s_{i+1} = H(v, i, s_i) \to w = s_n))$$

que es Π^0_1.

Para ello hay que probar primero que, fijado $v \in X$,

$$\bigwedge n \bigvee s(\ell(s) = n+1 \wedge s_0 = G(v) \wedge \bigwedge i < n\, s_{i+1} = H(v, i, s_i)).$$

Esto se prueba trivialmente por Σ^0_1-inducción, y de ahí se sigue que, dados $v \in X$ y $n \in \mathbb{N}$, existen w y s tales que $u = \langle\langle v, n\rangle_2, w\rangle_2$ cumple la definición de F (tomando $w = s_n$ en el resultado anterior). Otra inducción prueba que s es único (si s' cumple lo mismo, se prueba que $\bigwedge i < n+1\, s_i = s'_i$), y de ahí la equivalencia entre las dos definiciones de F alternativas.

En efecto, si u cumple la definición Π^0_1, tenemos que $u = \langle\langle v, n\rangle_2, w\rangle_2$, con $v \in X$. Entonces sabemos que existe s' tal que $u' = \langle\langle v, n\rangle_2, s'_n\rangle_2$ cumple la definición Σ^0_1 y, aplicando a este s' la definición Π^0_1, concluimos que $s'_n = w$, por lo que u cumple la definición Σ^0_1.

Recíprocamente, si u cumple la definición Σ^0_1 con s y suponemos que s' cumple la hipótesis de la definición Π^0_1, la unicidad hace que $s' = s$, luego $w = s_n = s'_n$ y así s' cumple lo requerido para que u cumpla la definición Π^0_1.

Esto nos permite concluir que $F : X \times \mathbb{N} \longrightarrow Y$, pues hemos visto que, dado $\langle v, n\rangle_2 \in X \times \mathbb{N}$, siempre existen s y w de modo que $u = \langle\langle v, n\rangle_2\, w\rangle_2$ cumple la definición de F, y w es único, con lo que $\langle\langle v, n\rangle_2, w\rangle_2$ es el único par en F con $\langle v, n\rangle_2$ como primera componente. Por lo tanto, $F : X \times \mathbb{N} \longrightarrow \mathbb{N}$ y $F(v) = w$. Más aún, una simple inducción prueba que $\bigwedge i < n\, F(v, i) = s_i$, lo que a su vez implica que F cumple lo requerido por el enunciado (en particular que toma imágenes en Y). ∎

Nota Modificando ligeramente la prueba del teorema anterior podemos demostrar también esta variante en la que no aparece X:

Si $H : \mathbb{N} \times Y \longrightarrow Y$ y $a \in Y$, existe una única función $F : \mathbb{N} \longrightarrow Y$ tal que

$$F(0) = a \wedge \bigwedge n\, F(n+1) = H(n, F(n)).$$

∎

Por ejemplo, dada una función $F : X \times \mathbb{N} \longrightarrow Y$, tomando $G(x) = 0$ y $H(x, n, y) = y^\frown F(x, n)$ obtenemos una función $F|_* : X \times \mathbb{N} \longrightarrow Y$ tal que

$$F|_0(x) = 0, \qquad F|_{n+1}(x) = F|_n(x)^\frown \langle F(x, n)\rangle\,.$$

[8]Notemos que la descripción $s_0 = G(v)$ es equivalente a $\bigvee u(u = \langle v, s_0\rangle_2 \wedge u \in G)$ y también a $\bigwedge u(u = \langle v, s_0\rangle_2 \to u \in G)$ y la fórmula $u = \langle v, s_0\rangle_2$ es Δ_1 en $\mathrm{I}\Sigma_1$, luego también en ACR_0, por lo que $s_0 = G(v)$ se puede sustituir por una fórmula Σ^0_1 y también por una fórmula Π^0_1. Lo mismo vale para $s_{i+1} = H(v, i, s_i)$ y otras fórmulas similares que nos van a aparecer.

Una simple inducción prueba que $\ell(F|_n(x)) = n$ y que

$$\bigwedge i < n\, (F|_n(x))_i = F(x, i).$$

Podemos expresar esto más claramente como que

$$F|_n(x) = \langle F(x,0), \ldots, F(x, n \dot{-} 1)\rangle\,.$$

A su vez, esto nos permite formular el teorema de recursión completa (compárese con el teorema 2.9):

Teorema 7.17 (Recursión completa) *Si $G : X \times \mathbb{N} \times \mathbb{N} \longrightarrow Y$, existe una única función $F : X \times \mathbb{N} \longrightarrow Y$ tal que*

$$\bigwedge x \in X \bigwedge n \in \mathbb{N}\, F(x,n) = G(x, n, F|_n(x)).$$

Demostración: Definimos $H : X \times \mathbb{N} \longrightarrow \mathbb{N}$ mediante

$$H(x,0) = 0, \qquad H(x, n+1) = H(x,n)^\frown\langle G(x, n, H(x,n))\rangle\,,$$

y a su vez $F(x,n) = H(x, n+1)_n$.

Una simple inducción muestra que $\ell(H(x,n)) = n$. A su vez, se cumple que $H(x,n) = F|_n(x)$. En efecto, para $n = 0$ es $H(x,0) = 0 = F|_0(x)$ y, si vale para n, usando las definiciones de H y F,

$$H(x, n+1) = H(x,n)^\frown\langle G(x,n,H(x,n))\rangle = F|_n(x)^\frown\langle H(x,n+1)_n\rangle =$$

$$F|_n(x)^\frown\langle F(x,n)\rangle = F|_{n+1}(x).$$

Por último probamos que F cumple lo requerido:

$$F(x,0) = H(x,1)_0 = G(x,0,H(x,0)) = G(x,0,F|_0(x)),$$

$$F(x,n+1) = H(x,n+2)_{n+1} = G(x,n+1,H(x,n+1)) = G(x,n+1,F|_{n+1}(x)).$$

Si dos aplicaciones F_1 y F_2 cumplen el teorema pero existen x, n tales que $F_1(x,n) \neq F_2(x,n)$, por 7.12 podemos tomar el mínimo n que cumple esto (para el x fijado), y entonces es fácil ver que $F_1|_n(x) = F_2|_n(x)$, de donde, aplicando G, concluimos que $F_1(x,n) = F_2(x,n)$ y tenemos una contradicción. ∎

También podemos definir funciones por minimización en el sentido siguiente:

Teorema 7.18 (Minimización) *Sea $G : X \times \mathbb{N} \longrightarrow \mathbb{N}$ una función tal que*

$$\bigwedge u \in X \bigvee v\, G(u,v) = 0.$$

Entonces existe una única función $F : X \longrightarrow \mathbb{N}$ tal que $F(x)$ es el mínimo número natural n que cumple $G(x,n) = 0$.

DEMOSTRACIÓN: Basta definir

$$F \equiv \{w \mid \bigvee uv \leq w(w = \langle u, v\rangle_2 \wedge u \in X \wedge G(u, v) = 0 \wedge \bigwedge i < v\, G(u, i) \neq 0)\}.$$

Como la definición es Δ_1^0, podemos aplicar el principio de Δ_1^0-especificación. El teorema 7.12 implica que

$$\bigwedge u \in X \overset{1}{\bigvee} v(G(u, v) = 0 \wedge \bigwedge i < v\, G(u, i) \neq 0),$$

de donde se sigue que $F : X \longrightarrow \mathbb{N}$. ∎

Diremos que la función F del teorema anterior es *la función definida por minimización* a partir de la función G o, equivalentemente, que es la dada por

$$F(x) = \mu v | G(x, v) = 0.$$

Relaciones Una *relación* R en un conjunto X es simplemente un subconjunto $R \subset X \times X$. En tal caso usamos la notación $x\, R\, y \equiv \langle x, y\rangle_2 \in R$.

Una relación $\leq$ en un conjunto X es una *relación de orden* si cumple las tres primeras propiedades de la lista siguiente (compárese con 5.67):

1. $\bigwedge x \in X\, x \leq x$,
2. $\bigwedge xy \in X(x \leq y \wedge y \leq x \to x = y)$,
3. $\bigwedge xyz \in X(x \leq y \wedge y \leq z \to x \leq z)$,
4. $\bigwedge xy \in X(x \leq y \vee y \leq x)$.

Si además cumple la cuarta propiedad, se dice que es una *relación de orden total.*

Por ejemplo, en el conjunto $\mathbb{N}$ tenemos definida la relación de orden usual

$$\leq \equiv \{w \mid \bigvee uv \leq w(w = \langle u, v\rangle_2 \wedge u \leq v)\},$$

de modo que $x \leq y$ puede interpretarse equivalentemente como la fórmula en la que $\leq$ es el relator diádico de $\mathcal{L}_a^2$ o bien como que $\langle x, y\rangle_2 \in \leq$. Esto nos permite particularizar a la relación de orden usual todos los resultados que probemos para relaciones de orden en general.

En general, cuando representamos por $\leq$ una relación de orden en un conjunto X, usamos la notación $x < y \equiv x \leq y \wedge x \neq y$.

Si R es una relación en un conjunto X e $Y \subset X$, entonces podemos considerar en Y la relación

$$R|_Y \equiv R \cap (Y \times Y).$$

Es claro que si R es una relación de orden (total) en X, entonces $R|_Y$ es una relación de orden (total) en Y. En general, cuando consideremos a un

conjunto X como conjunto ordenado con una relación $\leq$, consideraremos implícitamente a todos sus subconjuntos $Y \subset X$ como conjuntos ordenados con la restricción de $\leq$ a Y.

Si $\leq$ es una relación de orden en un conjunto X y tenemos un subconjunto $Y \subset X$, se dice que $a \in X$ es:

1. El *máximo* de Y si $a \in Y \wedge \bigwedge y \in X\, y \leq a$ (y es fácil ver que si Y tiene máximo, éste es único).
2. El *mínimo* de Y si $a \in Y \wedge \bigwedge y \in X\, a \leq y$ (y es fácil ver que si Y tiene mínimo, éste es único).
3. Un *maximal* de Y si $a \in Y \wedge \bigwedge y \in Y(a \leq y \to a = y)$.
4. Un *minimal* de Y si $a \in Y \wedge \bigwedge y \in Y(y \leq a \to y = a)$.

Es fácil ver que si Y está totalmente ordenado, todo maximal de Y es un máximo y todo minimal es un mínimo.

Si $\leq_X, \leq_Y$ son relaciones de orden en dos conjuntos X, Y, respectivamente, diremos que $F : (X, \leq_X) \longrightarrow (Y, \leq_Y)$ es *isótona* si $F : X \longrightarrow Y$ y

$$\bigwedge uv \in X(u \leq_X v \to F(u) \leq_Y F(v)).$$

Diremos que F es una *semejanza* si además $F : X \longrightarrow Y$ es biyectiva y $F^{-1} : (Y, \leq_Y) \longrightarrow (X, \leq_X)$ también es isótona.

Es fácil ver que la última condición se cumple trivialmente si las relaciones son de orden total.

Se dice que una relación de orden $\leq$ en un conjunto X es un *buen orden* en X si todo subconjunto no vacío de X tiene un mínimo elemento.

Notemos que todo buen orden es un orden total, pues si $x, y \in X$, el conjunto $\{x, y\}$ tiene que tener un mínimo elemento, y esto hace que $x \leq y$ o $y \leq x$, según si el mínimo es x o y.

Por ejemplo, el orden usual en $\mathbb{N}$ es un buen orden, pues si $Y \subset \mathbb{N}$ no es vacío, basta aplicar el teorema 7.12 a la fórmula $y \in Y$, de tipo Σ^0_1.

Teorema 7.19 *Si $\leq$ es un buen orden en X y $F : (X, \leq) \longrightarrow (X, \leq)$ es una aplicación inyectiva e isótona, entonces $\bigwedge x \in X\, x \leq F(x)$.*

Demostración: Supongamos que $\bigvee x \in X\, F(x) < x$. Podemos definir el conjunto

$$Y = \{x \mid x \in X \wedge F(x) < x\},$$

pues la fórmula que lo define es Σ^0_1:

$$x \in X \wedge F(x) < x \leftrightarrow x \in X \wedge \bigvee uwz(w = \langle x, u\rangle_2 \wedge z = \langle u, x\rangle \wedge w \in F \wedge z \in \leq)$$
$$\leftrightarrow \bigwedge uwz(w = \langle x, u\rangle_2 \wedge z = \langle u, x\rangle \wedge w \in F \to z \in \leq).$$

Por lo tanto, Y tiene un mínimo elemento x, que cumple $F(x) < x$, pero entonces $F(F(x)) < F(x)$, pero esto significa que $y = F(x) \in Y$, con $y < x$, en contradicción con la definición de mínimo. ∎

Conjuntos finitos Definimos así los conjuntos *finitos:*

$$X \text{ es finito} \equiv \bigvee k\ X \subset I_k \equiv \bigvee k \bigwedge i(i \in X \to i < k).$$

Vamos a probar que estos conjuntos finitos se corresponden con los conjuntos definidos aritméticamente en $\mathrm{I}\Sigma_1$. En el teorema siguiente, $i \in u$ es la relación de pertenencia definida en $\mathrm{I}\Sigma_1$:

Teorema 7.20 $\bigwedge X(X \textit{ es finito} \to \bigvee u \bigwedge i(i \in u \leftrightarrow i \in X))$.

Demostración: Supongamos que X es finito y fijemos k tal que $X \subset I_k$. Consideramos la fórmula de tipo Σ_1^0

$$\phi(j) \equiv j \leq k \to \bigvee u \bigwedge i < k(i \in u \leftrightarrow i \in X \wedge i < j).$$

Vamos a probar por inducción sobre j que $\bigwedge j\, \phi(j)$. Para $j = 0$ basta tomar $u = 0$. Supuesto $\phi(j)$, tenemos un número natural u tal que

$$\bigwedge i < k(i \in u \leftrightarrow i \in X \wedge i < j).$$

Supongamos ahora que $j + 1 \leq k$. Si $j \notin X$, entonces u hace que se cumpla $\phi(j + 1)$, mientras que si $j \in X$ tomamos $u' = u \cup \{j\}$ (donde la unión es la definida en $\mathrm{I}\Sigma_1$), que atestigua $\phi(j+1)$. De $\phi(k)$ deducimos la existencia de un u que cumple el teorema. ∎

La extensionalidad de la relación de pertenencia de $\mathrm{I}\Sigma_1$ implica que el conjunto u dado por el teorema anterior es único, por lo que si definimos el *código* de un conjunto finito como

$$[X] \equiv u \mid \bigwedge i(i \in u \leftrightarrow i \in X)$$

tenemos que

$$X \text{ es finito} \to \bigwedge i(i \in [X] \leftrightarrow i \in X),$$

donde el primer $\in$ es la pertenencia aritmética y el segundo la pertenencia de segundo orden.

Recíprocamente, todo número natural x es el código de un conjunto finito (único por el axioma de extensionalidad): $A_x \equiv \{u \mid u \in x\}$.

Teorema 7.21 $\bigwedge X(X \textit{ es finito} \leftrightarrow \bigvee Fn(F : I_n \longrightarrow X \textit{ biyectiva}))$.

Demostración: Si $F : I_n \longrightarrow X$ es biyectiva, vamos a demostrar por inducción sobre j que

$$j \leq n \to \bigvee k \bigwedge i < jF(i) < k.$$

Para $j = 0$ es trivial y basta tomar $k = 0$. Si es cierto para j y se cumple que $j + 1 \leq n$, por hipótesis de inducción tenemos un k tal que $\bigwedge i < j\, F(i) < k$. Entonces $k' = \max\{k, F(j)\}$ cumple lo requerido para $j+1$. En particular existe un k tal que $\bigwedge i < n\, F(i) < k$, pero esto implica que $X \subset I_k$, luego X es finito.

Recíprocamente, si X es finito, por 2.20 tenemos que existe un n y una aplicación $f : [I_n] \longrightarrow [X]$ biyectiva en el sentido aritmético (notemos que $[I_n]$ es el conjunto que en 2.20 llamamos I_n). Basta definir $F = \{u \mid u \in f\}$, y claramente se cumple que $F : I_n \longrightarrow X$ biyectiva. ∎

Un poco más en general:

Teorema 7.22 *Si $F : X \longrightarrow Y$ es una aplicación entre dos conjuntos finitos, entonces F es un conjunto finito y $[F] : [X] \longrightarrow [Y]$ es una aplicación en el sentido aritmético. Además $[F]$ es inyectiva, suprayectiva o biyectiva si y sólo si F lo es. Recíprocamente, si $f : [X] \longrightarrow [Y]$ es una aplicación en sentido aritmético, entonces $A_f : X \longrightarrow Y$ y $[A_f] = f$.*

Demostración: Podemos definir una aplicación $G : X \longrightarrow F$ biyectiva (dada por $G(x) = \langle x, F(x)\rangle_2$), y al componerla con una biyección $I_n \longrightarrow X$ tenemos una biyección $I_n \longrightarrow F$ que prueba que F es finito. El resto del teorema es inmediato. ∎

Como consecuencia, si X es un conjunto finito, existe un único n tal que existe $F : I_n \longrightarrow X$ biyectiva, pues si hubiera otro m con $G : I_m \longrightarrow X$ biyectiva, entonces $F \circ G^{-1} : I_n \longrightarrow I_m$ biyectiva, luego $[F \circ G^{-1}] : [I_n] \longrightarrow [I_m]$ biyectiva, y el teorema 2.21 implica que $m = n$.

Por lo tanto, podemos definir el cardinal de un conjunto finito como

$$|X| \equiv n \mid \bigvee F\, F : I_n \longrightarrow X \text{ biyectiva},$$

y los razonamientos precedentes muestran que, si X es finito, el cardinal $|X|$ de un conjunto finito X es el cardinal de su código $[X]$ en el sentido aritmético.

Las propiedades de los conjuntos finitos se pueden demostrar a través de sus códigos (lo que a menudo es necesario para evitar inducciones respecto de fórmulas de segundo orden). Por ejemplo:

$$\bigwedge XY(X, Y \text{ finitos} \wedge X \cap Y = \varnothing \to |X \cup Y| = |X| + |Y|),$$

$$\bigwedge XY(X, Y \text{ finitos} \to |X \times Y| = |X| \cdot |Y|),$$

En efecto, si X, Y son dos conjuntos finitos disjuntos, también $[X]$, $[Y]$ son conjuntos disjuntos en el sentido aritmético, y $[X \cup Y] = [X] \cup [Y]$ (donde la segunda unión es la definida aritméticamente). Entonces el teorema 5.56 nos da que $|[X] \cup [Y]| = |[X]| + |[Y]|$, luego $|X \cup Y| = |X| + |Y|$. Análogamente se razona con el producto cartesiano.

Similarmente se prueban los teoremas análogos a 5.55 o a 5.68 o a 5.69.

Enumeración de conjuntos infinitos Observemos que, por definición, un conjunto X es infinito si

$$\bigwedge u \bigvee v(u \leq v \wedge v \in X).$$

Por el teorema 7.12, si X es infinito tenemos, más precisamente, que

$$\bigwedge u \overset{1}{\bigvee} v(u \leq v \wedge v \in X \wedge \bigwedge i(u \leq i < v \to i \notin X)).$$

Esto nos lleva a definir

$$\mu_X \equiv \{w \mid \bigvee uv \leq w(w = \langle u, v\rangle_2 \wedge u \leq v \wedge$$

$$v \in X \wedge \bigwedge i < v(u \leq i \to i \notin X))\}.$$

La fórmula es claramente Δ_0^0, luego, si X es infinito, $\mu_X : \mathbb{N} \longrightarrow \mathbb{N}$ y cumple

$$\bigwedge u(\mu_X(u) \in X \wedge \bigwedge v(u \leq v < \mu_X(u) \to v \notin X)).$$

En otras palabras, $\mu_X(x)$ es el menor elemento de X mayor o igual que x. A su vez, la nota tras 7.16 nos permite definir la función π_X dada por

$$\pi_X(0) = \mu_X(0) \wedge \bigwedge n\ \pi_X(n+1) = \mu_X(\pi_X(n)+1).$$

Ahora es fácil probar que π_X es una enumeración creciente de X, es decir:

Teorema 7.23 *Si X es infinito, entonces $\pi_X : \mathbb{N} \longrightarrow \mathbb{N}$ y además:*

$$\bigwedge uv(u < v \to \pi_X(u) < \pi_X(v)), \qquad \bigwedge v(v \in X \leftrightarrow \bigvee u\, \pi_X(u) = v).$$

Demostración: Fijado x, demostramos $x < y \to \pi_X(x) < \pi_X(y)$ por inducción sobre y. Para $y = 0$ es trivial. Si vale para y y suponemos que $x < y+1$, entonces $x \leq y$. Si es $x < y$, entonces por hipótesis de inducción $\pi_X(x) < \pi_X(y) < \pi_X(y)+1 \leq \mu_X(\pi_X(y)+1) = \pi_X(y+1)$, luego $y+1$ cumple lo requerido. Si es $x = y$, entonces $\pi_X(x) = \pi_X(y) < \pi_X(y)+1$ y el razonamiento continúa como en el caso anterior.

Esto prueba la primera propiedad del enunciado, que equivale a que π_X es inyectiva e isótona, por lo que el teorema 7.19 nos da que $\bigwedge u\, u \leq \pi_X(u)$

Si $\pi_X(x) = y$, o bien $x = 0$, en cuyo caso $y = \pi_X(0) = \mu_X(0) \in X$, o bien $x = n+1$, en cuyo caso $y = \pi_X(n+1) = \mu_X(\pi_X(n)+1) \in X$, luego en cualquier caso $\bigvee u\ \pi_X(u) = y \to y \in X$.

Recíprocamente, si $y \in X$, tenemos que $y < y+1 \leq \pi_X(y+1)$, luego $\bigvee v\, y < \pi_X(v)$. Por 7.12, existe un z tal que $y < \pi_X(z) \wedge \bigwedge n < z\, \pi_X(n) \leq y$. No puede ser $z = 0$, pues en tal caso $\pi_X(z)$ sería el mínimo de X, y no puede ser mayor que $y \in X$. Por lo tanto $z = n+1$, luego

$$\pi_X(n) \leq y < \pi_X(n+1) = \mu_X(\pi_X(n)+1).$$

Si fuera $\pi_X(n) < y$, entonces $\pi_X(n)+1 \leq y \in X$, luego $\mu_X(\pi_X(n)+1) \leq y$, y tenemos una contradicción, luego tiene que ser $y = \pi_X(n)$. ∎

Veamos ahora que podemos probar un resultado ligeramente más fuerte:

Teorema 7.24 *Si $\phi(x)$ es una fórmula de tipo Σ^0_1 (tal vez con más variables libres), o bien existe un conjunto finito X tal que $\bigwedge n(n \in X \leftrightarrow \phi(n))$, o bien existe una función $F : \mathbb{N} \longrightarrow \mathbb{N}$ inyectiva tal que*

$$\bigwedge v(\phi(v) \leftrightarrow \bigvee u\, F(u) = v).$$

Demostración: Pongamos que $\phi(x) \equiv \bigvee i\, \psi(i, x)$, donde ψ es de tipo Δ^0_0. Sea

$$Y = \{w \mid \bigvee in \le w(w = \langle i, n\rangle_2 \wedge \psi(i, n) \wedge \bigwedge j < i \neg\psi(j, n))\}.$$

Así, si se cumple $\phi(x)$, tenemos que $\bigvee i\, \psi(i, x)$, luego por 7.12 existe un i tal que $\psi(i, x) \wedge \bigwedge j < i\, \neg\psi(j, x)$, luego $\langle i, x\rangle_2 \in Y$ y, recíprocamente, si $\bigvee i\ \langle i, x\rangle_2 \in Y$ entonces $\phi(x)$.

Si Y es finito, existe un k tal que $Y \subset I_k$, con lo que podemos definir

$$X = \{n \mid \bigvee i < k\ \langle i, n\rangle_2 \in Y\} \subset I_k,$$

que es un conjunto finito tal que $\bigwedge n(n \in X \leftrightarrow \phi(n))$. Supongamos ahora que Y es infinito, con lo que podemos considerar la función $\pi_Y : \mathbb{N}^1 \longrightarrow \mathbb{N}$ que enumera sus elementos. Basta definir F como la composición de π_Y con la proyección $\pi_2 : \mathbb{N} \longrightarrow \mathbb{N}$ dada por

$$\pi_2 \equiv \{w \mid \bigvee uv \le w(w = \langle u, v\rangle_2 \wedge v = u_2)\}.$$

Así, si se cumple $\phi(x)$, existe un i tal que $\langle i, x\rangle_2 \in Y$, luego existe un n tal que $\pi_Y(n) = \langle i, x\rangle_2$, y entonces $F(n) = \pi_2(\pi_X(n)) = x$. Recíprocamente, si $F(n) = x$, tenemos que $x = \pi_2(\pi_Y(n))$, luego $\pi_Y(n) = \langle i, x\rangle_2 \in Y$, luego $\phi(x)$.

Además F es inyectiva, pues si dos números cumplen $F(n) = F(m) = x$, entonces $x = \pi_2(\pi_Y(n)) = \pi_2(\pi_Y(m))$, luego $\pi_Y(n) = \langle i, x\rangle_2$ y $\pi_Y(m) = \langle j, x\rangle_2$, pero la definición de i implica que $i = j$, luego $\pi_Y(n) = \pi_Y(m)$, luego $n = m$. ∎

Rangos en $\mathbf{ACR_0}$ De aquí se deduce que en ACR_0 no se puede probar la existencia del rango de una función. Si pudiéramos demostrarla, el teorema anterior nos daría el principio de Σ^0_1-comprensión, y el teorema 7.9 nos daría que ACR_0 es la misma teoría que ACA_0, cosa que veremos que no es cierta. ∎

Definición 7.25 El *principio de Σ^0_n-especificación acotada* es el esquema

$$\bigwedge \bar{V} \bigwedge \bar{v} \bigwedge n \bigvee U \bigwedge i(i \in U \leftrightarrow i < n \wedge \phi(i)),$$

para toda fórmula $\phi(x, \bar{y}, \bar{Y})$ de tipo Σ^0_n. Análogamente se define el *principio de Π^0_1-especificación acotada.*

En otras palabras, el principio de Σ^0_n-especificación acotada afirma que las fórmulas de tipo Σ^0_n definen subconjuntos de cada conjunto I_n (y es fácil ver que, en consecuencia, definen subconjuntos de cada conjunto finito).

Teorema 7.26 *En* ACR_0 *se demuestra el principio de* Σ^0_1*-especificación acotada.*

DEMOSTRACIÓN: Sea $\phi(x)$ una fórmula de tipo Σ^0_1 y supongamos que, fijado un n,

$$\neg\bigvee U\bigwedge i(i \in U \leftrightarrow i < n \wedge \phi(i)).$$

Observemos que un conjunto U que cumpliera eso cumpliría $U \subset I_n$, luego sería finito. Por ello, podemos afirmar que no existe ningún conjunto finito U que cumpla lo indicado. El teorema 7.24 aplicado a la fórmula $x < n \wedge \phi(x)$ nos da la existencia de una función $F : \mathbb{N}^1 \longrightarrow \mathbb{N}$ inyectiva tal que $\bigwedge u(F(u) < n \wedge \phi(u))$. En particular, $F : \mathbb{N}^1 \longrightarrow I_n$ inyectiva, que a su vez se restringe a una aplicación $G : I_{n+1} \longrightarrow I_n$ inyectiva, pero esto implica que $n + 1 = |I_{n+1}| \leq |I_n| = n$, y tenemos una contradicción. ∎

Usaremos la notación

$$\{i < n \mid \phi(i)\} \equiv U|\bigwedge i(i \in U \leftrightarrow i < n \wedge \phi(i))$$

para denotar los conjuntos definidos por el principio de Σ^0_1-especificación acotada, que están bien definidos siempre que ϕ es una fórmula de tipo Σ^0_1, y de hecho también si es de tipo Π^0_1, pues

$$x \in I_n \setminus \{i < n \mid \neg\phi(i)\} \leftrightarrow x < n \wedge \phi(x),$$

de modo que si ϕ es de tipo Π^0_1, entonces $\neg\pi$ es de tipo Σ^0_1, con lo que el conjunto de la izquierda está bien definido. Por lo tanto:

Teorema 7.27 *En* ACR_0 *se demuestra el principio de* Π^0_1*-especificación acotada.*

Nota El axioma de inducción

$$\bigwedge U(0 \in U \wedge \bigwedge u(u \in U \to u' \in U) \to \bigwedge u\, u \in U)$$

es un caso particular del principio de Σ^0_1-inducción de ACR_0, aplicado a la fórmula $\alpha(x) \equiv x \in Y$. Combinado con el axioma de comprensión de AP^2, o con el principio de comprensión aritmética de ACA_0 permite demostrar principios de inducción más fuertes que el principio de Σ^0_1-inducción, pero, si contamos únicamente con el principio de Δ^0_1-comprensión de ACR_0, resulta ser más débil que el principio de Σ^0_1-inducción.

Sin embargo, sucede que si en ACR_0 sustituimos el principio de Σ^0_1-inducción por el axioma de inducción, restringimos el principio de Δ^0_1 comprensión a fórmulas Δ^0_0 y añadimos como esquema axiomático el principio de Σ^0_1-especificación acotada, entonces podemos demostrar el principio de Σ^0_1-inducción, con lo que obtenemos una axiomatización equivalente.

En efecto, fijamos una fórmula $\alpha(x, \bar{y}, \bar{Y})$ de tipo Σ^0_1 y vamos a demostrar

$$\bigwedge \bar{V}\bigwedge \bar{v}(\alpha(0) \wedge \bigwedge u(\alpha(u) \to \alpha(u')) \to \bigwedge u\, \alpha(u)).$$

Para ello fijamos unos parámetros $\bar{V}$ y $\bar{v}$, suponemos $\alpha(0) \wedge \bigwedge u(\alpha(u) \to \alpha(u'))$ y fijamos un número natural n. El principio de Σ^0_1-comprensión acotada (aplicado a $n+1$) nos da un conjunto X tal que

$$\bigwedge u(u \in X \leftrightarrow u \le n \wedge \alpha(u)).$$

Ahora aplicamos el principio de Δ^0_0-comprensión para obtener un conjunto Y tal que

$$\bigwedge u(u \in Y \leftrightarrow u \in X \vee u > n).$$

Así tenemos $0 \in Y \wedge \bigwedge u(u \in Y \to u' \in Y)$, luego el principio de inducción nos da que $\bigwedge u\, u \in Y$. En particular, $n \in Y$, luego $n \in X$, luego $\alpha(n)$, y con esto hemos probado que $\bigwedge n\, \alpha(n)$. ∎

Terminamos probando el teorema siguiente:

Teorema 7.28 *La teoría* $\mathrm{ACR}_0 + \Sigma^0_n$*-inducción es finitamente axiomatizable.*

Demostración: La prueba del teorema 6.11 se puede modificar trivialmente para probar en ACR_0 la existencia de una fórmula $\mathbb{N} \vDash^0_0 \alpha[v][V]$ de tipo Δ^0_1 con tres variables libres, α, v y V de modo que si α es una semifórmula de $\mathcal{L}^2_a$ de tipo Δ^0_0, v es una aplicación (en el sentido aritmético) definida al menos sobre las variables libres en α y V es un conjunto cualquiera, entonces se cumplen las mismas propiedades indicadas en el teorema y ésta otra:

Si $\alpha \equiv X_i(t)$*, entonces* $\mathbb{N} \vDash^0_0 \alpha[v][V] \leftrightarrow \mathrm{Dn}(t, v) \in V_i$.

A su vez, la definición 6.13 se adapta para definir fórmulas

$$\mathbb{N} \vDash_{\Sigma^0_n} \alpha[v][V], \qquad \mathbb{N} \vDash_{\Pi^0_n} \alpha[v][V],$$

de tipo Σ^0_n y Π^0_n, respectivamente que satisfacen el esquema teoremático análogo a 6.15:

Si $\phi(u_0, \ldots, u_{r\dot{-}1}, U_0, \ldots, U_{s\dot{-}1})$ *es una semifórmula de* $\mathcal{L}^2_a$ *de tipo* Σ^0_n *cuyas variables libres (todas ligadas) están entre las indicadas, en* ACR_0 *se demuestra:*

$$\bigwedge \bar{u}\bar{U}(\phi(\bar{u}, \bar{U}) \leftrightarrow \mathbb{N} \vDash_{\Sigma^0_n} \ulcorner\phi\urcorner[\{(\ulcorner u_i\urcorner, u_i)\}][\textstyle\bigcup_i(\{i\} \times U_i)]),$$

donde hemos abreviado:

$$\{(\ulcorner u_i\urcorner, u_i)\} \equiv \{(\ulcorner u_1\urcorner, u_1), \ldots, (\ulcorner u_r\urcorner, u_r)\},$$

$$\textstyle\bigcup_i(\{i\} \times U_i) \equiv (\{0\} \times U_0) \cup \cdots \cup (\{r \dot{-} 1\} \times U_{r\dot{-}1}),$$

y lo mismo vale cambiando Σ^0_n *por* Π^0_n.

Y, como en el caso de $\mathrm{I}\Sigma_1$, podemos probar que basta un número finito de axiomas de ACR_0 para demostrar todos los casos particulares de este esquema teoremático. (Entre dichos axiomas podemos tomar los necesarios para demostrar todos los teoremas de $\mathrm{I}\Sigma_1$, lo que nos asegura que contamos con todos los casos particulares del teorema 6.14, luego la prueba se reduce a generalizar de forma obvia las observaciones sobre el teorema 6.15 previas al teorema 6.19.)

Finalmente, el mismo argumento empleado en la prueba de 6.19 nos da que todos los casos particulares del esquema de Σ^0_n-inducción pueden demostrarse a partir de uno de ellos, el correspondiente a la fórmula $\mathbb{N} \vDash_{\Sigma^0_n} \alpha[v^0_x][X]$.

Del mismo modo, todos los casos particulares del axioma de Δ^0_1-comprensión pueden probarse a partir de

$$\bigwedge u(\mathbb{N} \vDash_{\Sigma^0_1} \alpha[v^u_x, X] \leftrightarrow \mathbb{N} \vDash_{\Pi^0_1} \beta[v^u_x, X]) \to \bigvee U \bigwedge u(u \in U \leftrightarrow \mathbb{N} \vDash_{\Sigma^0_1} \alpha[v^u_x, X]),$$

pues el axioma correspondiente a las fórmulas α y β se obtiene sustituyendo en éste las variables correspondientes por $\ulcorner\alpha\urcorner$ y $\ulcorner\beta\urcorner$. ∎

7.4 El teorema de completitud semántica

Tal y como explicamos en la sección 6.6, al considerar en $\mathrm{I}\Sigma_1$ la definición de lenguaje formal dada en la sección 3.2 para ARP, es razonable restringirnos a lenguajes recursivos, es decir, lenguajes definibles mediante fórmulas de tipo Δ_1 en $\mathrm{I}\Sigma_1$. Aquí vamos a estudiar los lenguajes formales trabajando en ACR_0, para lo que podríamos exigir que las fórmulas que los definen sean de tipo Δ^0_1, pero, mejor aún, en virtud del principio de Δ^0_1-comprensión, podemos considerar equivalentemente lenguajes formales $\mathcal{L}$ definidos mediante conjuntos. Concretamente, a través de unos conjuntos

$$\mathrm{VarLib}(\mathcal{L}), \quad \mathrm{VarLig}(\mathcal{L}), \quad \mathrm{Const}(\mathcal{L}), \quad \mathrm{Rel}(\mathcal{L}), \quad \mathrm{Fun}(\mathcal{L}),$$

y por una quíntupla $\langle \neg, \vee, \bigwedge, \bigvee, | \rangle$ (donde prescindimos desde el principio de los conectores que consideramos definidos, pero mantenemos la posibilidad de prescindir del descriptor), así como de dos funciones

$$\mathrm{ind} : \mathrm{VarLib}(\mathcal{L}) \cup \mathrm{VarLig}(\mathcal{L}) \cup \mathrm{Const}(\mathcal{L}) \cup \mathrm{Rel}(\mathcal{L}) \cup \mathrm{Fun}(\mathcal{L}) \longrightarrow \mathbb{N},$$

$$\mathrm{rang} : \mathrm{Rel}(\mathcal{L}) \cup \mathrm{Fun}(\mathcal{L}) \longrightarrow \mathbb{N}.$$

A partir de estos elementos, que constituyen la definición de $\mathcal{L}$, las fórmulas definidas en ARP definen en ACR_0 los conjuntos $\mathrm{SExp}(\mathcal{L})$, $\mathrm{STerm}(\mathcal{L})$ y $\mathrm{SForm}(\mathcal{L})$ de las semiexpresiones, semitérminos y semifórmulas de $\mathcal{L}$, y a su vez los conjuntos $\mathrm{Exp}(\mathcal{L})$, $\mathrm{Term}(\mathcal{L})$ y $\mathrm{Form}(\mathcal{L})$ de las expresiones, términos y fórmulas, y también los conjuntos $\mathrm{Des}(\mathcal{L})$ y $\mathrm{Sent}(\mathcal{L})$ de designadores y sentencias de $\mathcal{L}$ (términos y fórmulas sin variables libres).

A su vez, tenemos definido el conjunto $\mathrm{AxL}(\mathcal{L})$ de los axiomas lógicos de $\mathcal{L}$, y la fórmula $\Gamma \overset{d}{\vdash} \alpha$, que expresa que d es una deducción de α con premisas en el conjunto $\Gamma \subset \mathrm{Form}(\mathcal{L})$.

En realidad podemos considerar dos versiones de esta fórmula, una es literalmente la traducción de la fórmula correspondiente de $\mathrm{I}\Sigma_1$, en la que Γ es una variable de primer orden, pero podemos modificarla ligeramente para admitir que Γ sea una variable de segundo orden, de modo que el conjunto de premisas puede ser infinito, pese a que en una deducción sólo podrá aparecer una cantidad finita de ellas. En tal caso $\Gamma \overset{d}{\vdash} \alpha$ es una fórmula de tipo Δ^0_1 en ACR_0.

En estos términos, una teoría axiomática T, en el sentido de 3.26, sobre un lenguaje formal $\mathcal{L}$ puede definirse simplemente como un conjunto de axiomas propios

$$\mathrm{Ax}(T) \subset \mathrm{Form}(\mathcal{L}),$$

de modo que, si $\Gamma \subset \mathrm{Form}(\mathcal{L})$, podemos considerar la fórmula $\Gamma \overset{d}{\underset{T}{\vdash}} \alpha$ que cumple

$$\Gamma \overset{d}{\underset{T}{\vdash}} \alpha \leftrightarrow \Gamma \cup \mathrm{Ax}(T) \overset{d}{\vdash} \alpha.$$

Nuevamente, podemos considerar varias variantes de esta fórmula según si hacemos que las variables Γ y $T \equiv \mathrm{Ax}(T)$ sean de primer o de segundo orden.

En particular, $K_{\mathcal{L}}$ es la teoría axiomática determinada por $\mathrm{Ax}(K_{\mathcal{L}}) = \varnothing$.

Naturalmente, podemos definir

$$\Gamma \underset{T}{\vdash} \alpha \equiv \bigvee d\ \Gamma \overset{d}{\underset{T}{\vdash}} \alpha,$$

que es una fórmula de tipo Σ^0_1, por lo que en ACR_0 no podemos asegurar que defina un conjunto. Así pues, podemos hablar de los teoremas de una teoría axiomática T, pero no del conjunto formado por todos ellos.

Modelos En 3.3 introdujimos informalmente el concepto de modelo de un lenguaje formal. Ahora estamos en condiciones de formalizarlo junto con los pocos resultados que probamos informalmente sobre modelos, pero además demostraremos otros mucho más relevantes.

Observemos que un conjunto M define una sucesión de conjuntos mediante

$$M_n = \{m \mid \langle m, n\rangle_2 \in M\}.$$

Definición 7.29 Si $\mathcal{L}$ es un lenguaje formal, un *premodelo* de $\mathcal{L}$ es un conjunto M tal que:

1. M_0 es un conjunto no vacío, llamado *universo* del premodelo. En la práctica escribiremos M en lugar de M_0.

2. $M_1 : \mathrm{Const}(\mathcal{L}) \longrightarrow M_0$. En la práctica escribiremos $M(c) \in M$ en lugar de $M_1(c)$.

   Si $\mathcal{L}$ tiene descriptor supondremos que M_1 está definida sobre el conjunto $\mathrm{Const}(\mathcal{L}) \cup \{|\}$, y a $d = M_1(|) \in M$ lo llamaremos *descripción impropia* de M.

3. $I = M_2$ es un conjunto tal que si $f \in \mathrm{Fun}(\mathcal{L})$ es un funtor n-ádico, entonces $I_f : M_0^n \longrightarrow M_0$. En la práctica escribiremos $M(f) = M_f : M^n \longrightarrow M$, y si $R \in \mathrm{Rel}(\mathcal{L})$ es un relator n-ádico, entonces $I_R \subset M_0^n$ es una relación n-ádica en M_0. En la práctica escribiremos $M(R) = M_R \subset M^n$.

   Además, se tiene que cumplir que

   $$M(=) = \{v \mid \bigvee u < v(u \in M_0 \wedge v = \langle u, u\rangle_2)\},$$

   es decir, que $M(=)$ es la relación de igualdad en M.

Esta definición formaliza completamente la definición informal de modelo dada en 3.3, pero, por los motivos que explicaremos enseguida, no nos sirve como definición de modelo si queremos trabajar en ACR_0.

Si M es un modelo de $\mathcal{L}$, diremos que v es una *valoración* de una semiexpresión θ en M si $v : d \longrightarrow r$ es una función en el sentido aritmético cuyo dominio d es un conjunto de variables de $\mathcal{L}$ que incluye a todas las variables (libres o ligadas) que están libres en θ y cuyo rango r está contenido en M.

Llamaremos P_M al conjunto de todos los pares $\langle \theta, v\rangle_2$, tales que v es una valoración en M de la semiexpresión θ de $\mathcal{L}$.

Diremos que M es un *modelo* de $\mathcal{L}$ si es un premodelo que además cumple que

$$M_3 : P_M \longrightarrow M \cup \{0, 1\}$$

es una aplicación tal que, si t es un semitérmino, $M_3(t, v) \in M$, si α es una semifórmula, $M_3(t, \theta) \in \{0, 1\}$, y además se cumplen las propiedades siguientes, donde escribimos

$M(t)[v] \equiv M_3(t, v)$ si $\langle t, v\rangle_2 \in P_M$ y t es un semitérmino, y

$M \vDash \alpha[v] \equiv M(\alpha, v) = 1$ si $\langle \alpha, v\rangle_2 \in P_M$ y α es una semifórmula.

1. Si x es una variable (libre o ligada) de $\mathcal{L}$, entonces $M(x)[v] = v(x)$.
2. Si c es una constante de $\mathcal{L}$, entonces $M(c)[v] = M(c)$.
3. Si f es un funtor n-ádico de $\mathcal{L}$ y $t_1, \ldots, t_n$ son semitérminos,

$$M(f(t_1, \ldots, t_n))[v] = M(f)(M(t_1)[v], \ldots, M(t_n)[v]).$$

4. Si R es un relator n-ádico de $\mathcal{L}$ y $t_1, \ldots, t_n$ son semitérminos, entonces

$$M \vDash R(t_1, \ldots, t_n)[v] \leftrightarrow M(R)(M(t_1)[v], \ldots, M(t_n)[v]).$$

5. Si α y β son semifórmulas de $\mathcal{L}$, entonces

$$M \vDash \neg\alpha[v] \leftrightarrow \neg M \vDash \alpha[v], \qquad M \vDash (\alpha \vee \beta)[v] \leftrightarrow M \vDash \alpha[v] \vee M \vDash \beta[v].$$

6. Si $\alpha(u)$ es una semifórmula, entonces

$$M \vDash \bigwedge u\, \alpha(u)[v] \leftrightarrow \bigwedge a \in M\, M \vDash \alpha[v^a_u],$$

$$M \vDash \bigvee u\, \alpha(u)[v] \leftrightarrow \bigvee a \in M\, M \vDash \alpha[v^a_u],$$

$$M(u|\alpha(u))[v] = \begin{cases} a & \text{si } a \text{ es el único } a \in M \text{ tal que } M \vDash \alpha[v^a_u], \\ d & \text{en caso contrario,} \end{cases}$$

donde $v^a_u \equiv (v \setminus \{\langle u, v(u)\rangle_2\}) \cup \{\langle u, a\rangle_2\}$ y d es la descripción impropia del premodelo M.

Una simple inducción prueba que la función M_3 de un modelo es única, en el sentido de que si dos funciones cumplen las propiedades requeridas, necesariamente son la misma, pero no podemos probar en ACR_0 que para todo premodelo exista una función M_3 que lo convierta en modelo. En otras palabras, la definición 3.6 no es formalizable en ACR_0 (véase el teorema 7.45 más adelante). Ahora bien, admitiendo que exista M_3, es decir, admitiendo que tenemos un modelo, los teoremas 3.9 y 3.12 se demuestran formalmente sin dificultad, por inducción sobre θ.

Diremos que una fórmula α es *verdadera* en M si

$$M \vDash \alpha \equiv \bigwedge v(\langle \alpha, v\rangle_2 \in P_M \to M \vDash \alpha[v]),$$

y diremos que α es *falsa* en M si

$$\bigwedge v(\langle \alpha, v\rangle_2 \in P_M \to \neg M \vDash \alpha[v]).$$

Si Γ es un conjunto de fórmulas de $\mathcal{L}$, se dice que un modelo M de $\mathcal{L}$ es un modelo de Γ si

$$M \vDash \Gamma \equiv \bigwedge \gamma \in \Gamma\, M \vDash \gamma.$$

Es pura rutina formalizar en ACR_0 el teorema de corrección 3.16. De hecho, una mínima variación de la prueba nos permite enunciarlo con conjuntos posiblemente infinitos:

Teorema 7.30 (Teorema de corrección) *Sea $\mathcal{L}$ un lenguaje formal y Γ un conjunto de fórmulas de $\mathcal{L}$ tales que $\Gamma \vdash \alpha$. Entonces, si $M \vDash \Gamma$, también se cumple $M \vDash \alpha$.*

Es fácil definir un premodelo M del lenguaje $\mathcal{L}_a$ de la aritmética de primer orden tomando como universo $M = \mathbb{N}$, interpretando la constante 0 como $M(0) = 0$ y los funtores suma y producto como la suma y el producto usuales de números naturales (y el relator de orden como la relación de orden usual). Sin embargo, en ACR_0 no podemos dotar a este premodelo de estructura de modelo, y así no podemos plantear siquiera en ACR_0 que M sea un modelo de la aritmética de Peano.

A lo sumo, no es difícil probar que todo premodelo de universo finito puede extenderse a un modelo, pero la utilidad de este hecho es moderada (con ello podemos probar al menos que determinadas fórmulas no son consecuencia de otras dadas en algunos casos sencillos). Veamos ahora qué podemos decir sobre la existencia de modelos en general. Para ello necesitamos introducir algunos conceptos.

Completitud Un conjunto $\Gamma \subset \mathrm{Form}(\mathcal{L})$ es *consistente* o *completo* si cumple, respectivamente

$$\mathrm{Consis}\,\Gamma \equiv \neg\bigvee\alpha\,(\Gamma \vdash \alpha \wedge \Gamma \vdash \neg\alpha),$$

$$\mathrm{Comp}\,\Gamma \equiv \bigwedge \alpha \in \mathrm{Sent}(\mathcal{L})(\Gamma \vdash \alpha \vee \Gamma \vdash \neg\alpha).$$

Notemos que en el caso de la completitud es fundamental restringir la definición a sentencias, pues en una teoría axiomática consistente no se puede demostrar $x \neq y$, pero no es razonable esperar que se pueda demostrar $x = y$.

Se dice que Γ es *contradictorio* si no es consistente, es decir, si a partir de Γ se puede demostrar una contradicción, pero en tal caso de Γ se deduce cualquier fórmula, luego Γ es consistente si y sólo si existe una fórmula no deducible de Γ, o también si $x \neq x$ (por ejemplo) no es deducible de Γ.

Es obvio que un conjunto de fórmulas Γ que admita un modelo M es consistente, pues si $M \models \Gamma$ y existiera una fórmula α tal que $\Gamma \vdash \alpha$ y $\Gamma \vdash \neg\alpha$, por el teorema de corrección $M \models \alpha$ y $M \models \neg\alpha$, pero esto equivale a que $M \models \alpha$ y $\neg M \models \alpha$, con lo que tenemos una contradicción.

Diremos que $\Gamma \subset \mathrm{Sent}(\mathcal{L})$ es un *conjunto consistente maximal* si es consistente y, para toda sentencia α de $\mathcal{L}$ tal que $\alpha \notin \Gamma$, se cumple que $\Gamma \cup \{\alpha\}$ es contradictorio.

Observemos que $\Gamma \subset \mathrm{Sent}(\mathcal{L})$ es un conjunto consistente maximal si y sólo si es consistente y, para toda sentencia α, se cumple que $\alpha \in \Gamma$ o $\neg\alpha \in \Gamma$ (pero no se dan ambos casos).

En efecto, si Γ es consistente maximal y $\alpha \notin \Gamma$, entonces $\Gamma \cup \{\alpha\}$ es contradictorio, luego $\Gamma \vdash \neg\alpha$ (porque suponiendo α se llega a una contradicción), luego $\Gamma \cup \{\neg\alpha\}$ es consistente (porque si se pudiera deducir una contradicción, deduciendo $\neg\alpha$ de Γ probaríamos la misma contradicción a partir de Γ, que es consistente), luego $\neg\alpha \in \Gamma$, y no se pueden dar los dos casos porque entonces Γ sería contradictorio.

Recíprocamente, si se cumple esta condición y $\alpha \notin \Gamma$, es que $\neg\alpha \in \Gamma$, luego $\Gamma \cup \{\alpha\}$ es contradictorio, luego Γ es consistente maximal.

Esto implica que Γ es cerrado para consecuencias lógicas, es decir, que si α es una sentencia tal que $\Gamma \vdash \alpha$, entonces $\alpha \in \Gamma$, pues en caso contrario $\neg\alpha \in \Gamma$ y Γ sería contradictorio.

Teorema 7.31 *Si Γ es un conjunto consistente maximal de sentencias de un lenguaje formal $\mathcal{L}$, entonces:*

1. $\neg\alpha \in \Gamma$ *si y sólo si* $\alpha \notin \Gamma$.
2. $\alpha \vee \beta \in \Gamma$ *si y sólo si* $\alpha \in \Gamma \vee \beta \in \Gamma$.
3. $\alpha \wedge \beta \in \Gamma$ *si y sólo si* $\alpha \in \Gamma \wedge \beta \in \Gamma$.
4. $\alpha \to \beta \in \Gamma$ *si y sólo si* $\alpha \notin \Gamma \vee \beta \in \Gamma$.
5. $\alpha \leftrightarrow \beta \in \Gamma$ *si y sólo si* $(\alpha \in \Gamma \wedge \beta \in \Gamma) \vee (\alpha \notin \Gamma \wedge \beta \notin \Gamma)$.

Demostración: La propiedad 1. ya la hemos demostrado y hemos visto que, de hecho, caracteriza la consistencia maximal.

2. Si $\alpha \in \Gamma \vee \beta \in \Gamma$, entonces $\Gamma \vdash \alpha \vee \beta$, luego $\alpha \vee \beta \in \Gamma$. Recíprocamente, si $\alpha \notin \Gamma \wedge \beta \notin \Gamma$, entonces $\neg\alpha \in \Gamma \wedge \neg\beta \in \Gamma$, luego tenemos que $\Gamma \vdash \neg(\alpha \vee \beta)$, luego $\neg(\alpha \vee \beta) \in \Gamma$, luego $\alpha \vee \beta \notin \Gamma$.

3. El hecho de que Γ sea cerrado para consecuencias lógicas nos da la primera de las implicaciones siguientes, y las restantes se deben a los casos anteriores:

$\alpha \wedge \beta \in \Gamma$ si y sólo si $\neg(\neg\alpha \vee \neg\beta) \in \Gamma$ si y sólo si $\neg\alpha \vee \neg\beta \notin \Gamma$ si y sólo si $\neg\alpha \notin \Gamma \wedge \neg\beta \notin \Gamma$ si y sólo si $\alpha \in \Gamma \wedge \beta \in \Gamma$.

Los casos restantes se prueban análogamente. ∎

Observemos que si M es un modelo de un lenguaje formal $\mathcal{L}$, entonces el conjunto

$$T(M) \equiv \{\alpha \mid \alpha \in \mathrm{Sent}(\mathcal{L}) \wedge M \vDash \alpha\}$$

es un conjunto consistente maximal de sentencias de $\mathcal{L}$ tal que $M \vDash T(M)$.

Vamos a ver que, recíprocamente, a partir de conjuntos consistentes maximales de sentencias de un lenguaje formal dado podemos construir modelos. Para ello necesitamos recordar el concepto de árbol:

Árboles En [CS 1.1] hemos introducido el concepto de árbol para definir las deducciones del cálculo secuencial. Repetimos aquí las definiciones básicas:

Definición 7.32 Un *árbol* es un conjunto A tal que $\bigwedge s \in A \bigwedge i \leq \ell(s)\, s|_i \in A$.

A los elementos de un árbol los llamaremos *nodos*. Es costumbre llamar *altura* de un nodo a su longitud. Todo árbol A es un conjunto parcialmente ordenado con el orden dado por

$$s \preceq t \leftrightarrow s = t|_{\ell(s)}.$$

Todo árbol no vacío tiene como mínimo elemento al nodo nulo 0.

Un árbol A es *finitamente ramificado* si cada nodo tiene un número finito de sucesores inmediatos, es decir, si

$$\bigwedge s(s \in A \to \bigvee n \bigwedge m(s^\frown\langle m\rangle \in A \to m \leq n)).$$

El *árbol binario completo* es

$$2^{<\omega} \equiv \{s \mid \bigwedge i < \ell(s)(s_i = 0 \vee s_i = 1)\},$$

que claramente es un árbol. Más en general, un *árbol binario* es un conjunto $A \subset 2^{<\omega}$ tal que $\bigwedge s \in A \bigwedge i \leq \ell(s)\, s|_i \in A$. Claramente, todo árbol binario es finitamente ramificado.

El *árbol binario completo de altura* n es el árbol

$$2^{<n} \equiv \{s \mid s \in 2^{<\omega} \wedge \ell(s) < n\}.$$

es fácil ver que es un conjunto finito.

Si $F : \mathbb{N} \longrightarrow \mathbb{N}$, entonces $F|_{I_n} : I_n \longrightarrow \mathbb{N}$ es un conjunto finito, cuyo código $[F|_{I_n}] : I_n \longrightarrow \mathbb{N}$ (definido tras 7.20) es una aplicación en el sentido aritmético, que a su vez se corresponde con una sucesión F^n tal que

$$\ell(F^n) = n \wedge \bigwedge i < n\, F^n_i = F(i).$$

Si A es un árbol, diremos que $F : \mathbb{N} \longrightarrow \mathbb{N}$ es un *camino* en A si

$$\bigwedge n\, F^n \in A.$$

Notemos que $\bigwedge mn(m < n \to F^m \prec F^n)$, pues se cumple que $F|_{I_m} = (F|_{I_n})|_{I_m}$, luego $[F|_{I_m}] = [F|_{I_n}]|_{I_m}$, luego $F^m = F^n|_m$, que equivale a $F^m \preceq F^n$, y la desigualdad es estricta porque las alturas son distintas.

Así pues, un camino en un árbol A determina una sucesión estrictamente creciente de nodos, uno de cada altura posible.

Finalmente, si $s \in 2^{<\omega}$, llamaremos $[s] \equiv \{i \mid i < \ell(s) \wedge s_i = 1\}$. ∎

Volviendo al problema de la construcción de conjuntos consistentes maximales, veamos que podemos asociar un árbol binario a cada conjunto de sentencias:

Si $\Gamma \subset \mathrm{Sent}(\mathcal{L})$, definimos

$$A(\Gamma,\mathcal{L}) \equiv \{s \mid s \in 2^{<\omega} \wedge \bigwedge \alpha < \ell(s)(s_\alpha = 1 \to \alpha \in \mathrm{Sent}(\mathcal{L})) \wedge$$

$$\bigwedge \alpha < \ell(s)(\alpha \in \Gamma \to s_\alpha = 1) \wedge$$

$$\bigwedge \alpha\beta < \ell(s)(\alpha \in \mathrm{Sent}(\mathcal{L}) \wedge \beta = \neg\alpha \to s_\alpha + s_\beta = 1) \wedge$$

$$\bigwedge d\alpha < \ell(s)(\alpha \in \mathrm{Sent}(\mathcal{L}) \wedge [s] \overset{d}{\underset{\Gamma}{\vdash}} \alpha \to s_\alpha = 1)\}.$$

Notemos en primer lugar que la definición es Δ^0_1, pues la fórmula es equivalente a cualquiera de las fórmulas[9]

$$\bigvee x(x = \ell(s) \wedge \cdots), \qquad \bigwedge x(x = \ell(s) \to \cdots)$$

donde los puntos suspensivos se sustituyen por la fórmula que hemos usado en la definición salvo que sustituimos todo cuantificador $\bigwedge \alpha < \ell(s)$ por $\bigwedge \alpha < x$.

En segundo lugar observamos que el conjunto $A(\Gamma,\mathcal{L})$ es trivialmente un árbol binario. Una vez hemos justificado su existencia analizando su definición formal, podemos expresarlo en términos más amistosos:

El árbol $A(\Gamma,\mathcal{L})$ está formado por todos los nodos s que cumplen:

[9]Notemos que no es necesario hacer nada con los conjuntos 2^ω, $\mathrm{Sent}(\mathcal{L})$, etc. que aparecen en la definición. Podemos sustituirlos por variables ligadas $U = 2^{<\omega}$, $V = \mathrm{Sent}(\mathcal{L})$, etc., aplicar el axioma de Δ^0_1-especificación y luego eliminar los cuantificadores sustituyendo U por $2^{<\omega}$, etc., y así obtenemos la existencia de $A(\Gamma,\mathcal{L})$. Lo que sí que hacía falta era justificar que los cuantificadores acotados por la descripción $\ell(s)$ pueden sustituirse por cuantificadores acotados por términos aritméticos (variables, concretamente).

1. s sólo toma el valor 1 sobre números naturales que, vistos como sucesiones, son sentencias de $\mathcal{L}$.
2. s toma el valor 1 sobre todas las sentencias de Γ que están en su dominio.
3. Si unas sentencias α y $\neg\alpha$ están ambas en su dominio, s toma el valor 1 exactamente sobre una de las dos.
4. Si una sentencia $\alpha < \ell(s)$ es deducible a partir de $\Gamma \cup [s]$ mediante una deducción $d < \ell(s)$, entonces $s_\alpha = 1$.

Veamos algunas propiedades de estos árboles:

- *Si Γ es contradictorio, entonces $A(\Gamma, \mathcal{L})$ es finito.*

  En efecto, si Γ es contradictorio existen deducciones d y d' que terminan con una sentencia α y con $\neg\alpha$, respectivamente. Sea n un número mayor que d y d'. Basta ver que $A(\Gamma, \mathcal{L}) \subset 2^{<n}$.

  Esto se debe a que, si existiera $s \in A(\Gamma, \mathcal{L})$ tal que $n \leq \ell(s)$, entonces s debería tomar el valor 1 tanto en α como en $\neg\alpha$, por la condición 4 de la definición, pero esto contradice a la condición 3.

- *Si Γ es consistente, entonces $A(\Gamma, \mathcal{L})$ tiene elementos de altura arbitrariamente grande, luego es infinito.*

  Fijamos un número natural n, tomamos $m = 1 + \operatorname{máx} 2^{<n}$ y definimos

$$A_n(\Gamma, \mathcal{L}) \equiv \{s < m \mid s \in 2^{<n} \wedge \bigwedge \alpha < \ell(s)(\alpha \in \operatorname{Sent}(\mathcal{L}) \wedge \neg\bigvee d\ [s|_\alpha] \cup \{\alpha\} \overset{d}{\underset{\Gamma}{\vdash}} x \neq x \to s_\alpha = 1)\}.$$

  Notemos que la fórmula que define a $A_n(\Gamma, \mathcal{L})$ es de tipo Σ^0_1, por lo que el principio de Σ^0_1-especificación acotada implica que la definición es correcta. La condición $s < m$ es redundante, pues ya está contenida en $s \in 2^{<n}$.

  Vamos a probar por Π^0_1-inducción la fórmula

$$\bigwedge i(i < n \to \bigvee s < m(s \in A_n(\Gamma, \mathcal{L}) \wedge \neg\bigvee d\ [s] \overset{d}{\underset{\Gamma}{\vdash}} x \neq x \wedge \ell(s) = i)).$$

  Notemos que la segunda condición equivale a que $\Gamma \cup [s]$ sea consistente.

  Para $i = 0$ basta tomar $s = 0$, y aquí usamos que Γ es consistente. Supongamos que la propiedad es cierta para i, así como que $i+1 < n$. Por hipótesis de inducción existe $s \in A_n(\Gamma, \mathcal{L})$ tal que $\Gamma \cup [s]$ es consistente y $\ell(s) = i$.

  Si i, considerado como sucesión finita, no es una sentencia de $\mathcal{L}$, basta definir $s' = s^\frown\langle 0\rangle$, de modo que $s'_i = 0$, y claramente $s' \in A_n(\Gamma, \mathcal{L})$, $\ell(s') = i + 1$ y, como $[s'] = [s]$, se cumple que $\Gamma \cup [s']$ es consistente.

  Supongamos ahora que $\beta \equiv i \in \operatorname{Sent}(\mathcal{L})$. Si $\Gamma \cup [s] \cup \{\beta\}$ es consistente, definimos $s' = s^\frown\langle 1\rangle$ y en caso contrario $s' = s^\frown\langle 0\rangle$. Así en el primer caso $[s'] = [s] \cup \{\beta\}$, mientras que en el segundo $[s'] = [s]$, luego en ambos casos $\ell(s') = i + 1$ y $\Gamma \cup [s']$ es consistente. Ahora es fácil concluir que $s' \in A_n(\Gamma, \mathcal{L})$.

En particular, si aplicamos esto a $n+1$, existe $s \in A_{n+1}(\Gamma, \mathcal{L})$ tal que $\ell(s) = n$ y $\Gamma \cup [s]$ es consistente. Basta probar que $s \in A(\Gamma, \mathcal{L})$. Obviamente cumple la condición 1. Si $\alpha < \ell(s)$ cumple $\alpha \in \Gamma$, entonces la consistencia de $\Gamma \cup [s]$ implica en particular la de $\Gamma \cup [s|_\alpha] \cup \{\alpha\}$, y como $s \in A_{n+1}(\Gamma, \mathcal{L})$, esto implica que $s_\alpha = 1$, como requiere la propiedad 2.

Para probar 3 suponemos que $\alpha, \neg\alpha < \ell(s)$ son sentencias de $\mathcal{L}$. Si se cumple $s_\alpha = 1$, entonces $s_{\neg\alpha} = 0$, o de lo contrario $\Gamma \cup [s]$ sería contradictorio. Si, por el contrario, $s_\alpha = 0$, por definición de $A_{n+1}(\Gamma, \mathcal{L})$, esto significa que $\Gamma \cup [s|_\alpha] \cup \{\alpha\}$ es contradictorio, luego $\Gamma \cup [s|_\alpha] \vdash \neg\alpha$ (por reducción al absurdo, pues suponiendo α llegamos a una contradicción), luego $\Gamma \cup [s|_{\neg\alpha}] \cup \{\neg\alpha\}$ es consistente, pues si con dichas premisas pudiéramos probar una contradicción, también la podríamos probar con $\Gamma \cup [s|_{\neg\alpha}]$ (pues $\neg\alpha$ es deducible), luego $\Gamma \cup [s]$ también sería contradictorio. Por definición de $A_{n+1}(\Gamma, \mathcal{L})$ concluimos que $s_{\neg\alpha} = 1$.

Por último, si $\alpha < \ell(s)$ es una sentencia tal que $\Gamma \cup [s] \vdash \alpha$, entonces $\Gamma \cup [s] \cup \{\alpha\}$ es consistente (pues si pudiéramos probar una contradicción de dichas premisas, también podríamos probarla a partir de $\Gamma \cup [s]$, ya que α es deducible), luego también lo es $\Gamma \cup [s|_\alpha]$ y, por definición de $A_{n+1}(\Gamma, \mathcal{L})$, entonces $s_\alpha = 1$.

Esto prueba que $s \in A(\Gamma, \mathcal{L})$, con lo que este árbol contiene nodos de altura arbitrariamente grande.

- *Si Γ es consistente, los caminos en $A(\Gamma, \mathcal{L})$ son las funciones características de los conjuntos consistentes maximales de sentencias de $\mathcal{L}$ que contienen a Γ, es decir, que si $F : \mathbb{N} \longrightarrow \{0, 1\}$ es un camino en $A(\Gamma, \mathcal{L})$, entonces $\Gamma^* \equiv \{\alpha \mid F(\alpha) = 1\}$ es un conjunto consistente maximal de sentencias de $\mathcal{L}$ tal que $\Gamma \subset \Gamma^*$, y cualquiera de ellos es de esta forma, para cierto camino F.*

  En efecto, si F es un camino, Γ^* es consistente, pues si a partir de Γ^* se dedujera una contradicción, existirían deducciones d y d' tales que

$$\Gamma^* \overset{d}{\vdash} \alpha, \qquad \Gamma^* \overset{d'}{\vdash} \neg\alpha,$$

  y podríamos tomar un $n > d, d'$, con lo que $s = F^n \in A(\Gamma, \mathcal{L})$ cumpliría que $[s]$ contiene todas las premisas de Γ^* usadas en ambas deducciones, luego $[s] \overset{d}{\vdash} \alpha$, $[s] \overset{d'}{\vdash} \neg\alpha$, luego $s_\alpha = s_{\neg\alpha} = 1$, en contradicción con la definición del árbol.

  Por otra parte, dada una sentencia α, tomamos n mayor que α y $\neg\alpha$, con lo que $s = F^n \in A(\Gamma, \mathcal{L})$ cumple que $\alpha, \neg\alpha < \ell(s)$, luego $s_\alpha + s_{\neg\alpha} = 1$, luego $\alpha \in \Gamma^*$ o $\neg\alpha \in \Gamma^*$, luego Γ^* es consistente maximal.

  Recíprocamente, si Γ^* es un conjunto consistente maximal de sentencias de $\mathcal{L}$ que contiene a Γ, podemos definir $F \equiv (\Gamma^* \times \{1\}) \cup ((\mathbb{N} \setminus \Gamma^*) \times \{0\})$, de modo que $F : \mathbb{N} \longrightarrow \{0, 1\}$. Basta probar que F es un camino en $A(\Gamma, \mathcal{L})$, pues entonces es claro que el conjunto Γ^* definido a partir de él es nuestro conjunto de partida.

Así pues, fijamos un número natural n y consideramos $s = F^n$. Tenemos que probar que $s \in A(\Gamma, \mathcal{L})$. La condición 1 se cumple obviamente, la 2 se cumple porque $\Gamma \subset \Gamma^*$, la 3 se cumple porque Γ^* es consistente maximal (luego contiene a α o a $\neg\alpha$, pero no a ambas, para toda sentencia α), y la condición 4 se cumple porque si $\Gamma \cup [s] \vdash \alpha$, entonces $\Gamma^* \vdash \alpha$, luego $\alpha \in \Gamma^*$, luego $s_\alpha = 1$. ∎

Quizá el lector crea que hemos demostrado que todo conjunto consistente de sentencias se puede extender a un conjunto consistente maximal (formando el árbol $A(\Gamma, \mathcal{L})$ y tomando un camino), pero no es cierto, porque en ACR_0 no puede demostrarse que todo árbol binario tenga un camino:

Definición 7.33 El *lema de König débil* es la sentencia:

LKD: *Todo árbol binario infinito tiene un camino.*

Veremos que LKD no puede demostrarse en ACR_0, por lo que llamamos LKD_0 a la teoría que resulta de añadir LKD a los axiomas de ACR_0.

También veremos (teorema 7.41) que LKD sí que es demostrable en ACA_0, por lo que todos los teoremas de LKD_0 lo son también de ACA_0.

Por otra parte, en [CS 2.34] probamos que los teoremas de primer orden de $\mathrm{LKD}_0 + \Sigma^0_n$-inducción son exactamente los mismos que los de $\mathrm{ACR}_0 + \Sigma^0_n$-inducción, así como que $\mathrm{LKD}_0 + \Sigma^0_n$-inducción es una extensión conservativa de $\mathrm{I}\Sigma_n$.

En otras palabras, el lema de König débil permite demostrar teoremas de segundo orden que no son demostrables en ACR_0, pero no aporta ningún teorema nuevo de primer orden, ni tampoco ningún teorema formulable en $\mathcal{L}_a$ que no sea demostrable en $\mathrm{I}\Sigma_1$. En particular, la consistencia de $\mathrm{I}\Sigma_1$ implica la de LKD_0.

Continuando con nuestro análisis de la existencia de conjuntos consistentes maximales, de momento hemos demostrado el teorema siguiente, pero no en ACR_0, sino en LKD_0:

Teorema 7.34 (LKD_0) (Lema de Lindenbaum) *Todo conjunto consistente de sentencias de un lenguaje formal de primer orden $\mathcal{L}$ puede extenderse a un conjunto consistente maximal.*

Vamos a refinar sustancialmente esta conclusión, para lo cual necesitamos un hecho general sobre la lógica de primer orden:

Teorema 7.35 *Sea $\mathcal{L}$ un lenguaje formal de primer orden y $\alpha(x)$ una fórmula y c una constante que no aparezca en α, de modo que $\vdash \alpha(c)$. Entonces también $\vdash \alpha(x)$.*

DEMOSTRACIÓN: Puesto que en $\alpha(c)$ no aparece la variable x, podemos considerar una demostración de $\alpha(c)$ en la que no aparezca la variable x, digamos $\alpha_1, \ldots, \alpha_n$, con $\alpha_n \equiv \alpha(c)$. Cada fórmula α_i puede verse como $\alpha_i(c)$, para cierta fórmula $\alpha_i(x)$ que resulta de sustituir cada aparición de c por la variable x.

Basta observar que entonces $\alpha_1(x), \ldots, \alpha_n(x)$ es una demostración de $\alpha(x)$, pues una comprobación rutinaria muestra que si $\alpha_i(c)$ es un axioma lógico, también lo es $\alpha_i(x)$, así como que si $\alpha_i(c)$ se deduce de fórmulas anteriores por una de las reglas de inferencia MP o IG, entonces $\alpha_i(x)$ se deduce de las fórmulas correspondientes por la misma regla.

Por ejemplo, en el caso de IG tenemos que existe un índice $j < i$ tal que $\alpha_j(c) \equiv \alpha_j(c, y)$, para cierta variable $y \not\equiv x$ (porque x no aparece en la demostración) y $\alpha_i(c) \equiv \bigwedge u\, \alpha_j(c, u)$. Entonces $\alpha_i(x) \equiv \bigwedge u\, \alpha_j(x, u)$, que es consecuencia por IG de $\alpha_j(x) \equiv \alpha_j(x, y)$. ∎

Como consecuencia demostramos que todo conjunto consistente de sentencias se puede ejemplificar sin perder la consistencia, es decir, que si tenemos una sentencia del tipo $\bigvee u\, \alpha(u)$, podemos añadir una sentencia que diga que una constante "nueva" es un ejemplo de objeto que cumple $\alpha(x)$:

Teorema 7.36 *Sea Γ un conjunto consistente de sentencias de un lenguaje formal $\mathcal{L}$, una de las cuales sea $\bigvee u\, \alpha(u)$. Sea c una constante de $\mathcal{L}$ que no aparezca en ninguna de las sentencias de Γ. Entonces $\Gamma \cup \{\alpha(c)\}$ también es consistente.*

DEMOSTRACIÓN: Supongamos que $\Gamma \cup \{\alpha(c)\}$ es contradictorio. Entonces $\Gamma \vdash \neg\alpha(c)$ (por reducción al absurdo, pues suponiendo $\alpha(c)$ podemos probar una contradicción). Sean $\gamma_1, \ldots, \gamma_n$ las premisas que de hecho aparecen en la deducción de $\neg\alpha(c)$ Tenemos, pues, que

$$\gamma_1 \wedge \cdots \wedge \gamma_n \vdash \neg\alpha(c).$$

Por el teorema de deducción:

$$\vdash \gamma_1 \wedge \cdots \wedge \gamma_n \to \neg\alpha(c).$$

Teniendo en cuenta que c no aparece en el antecedente de la implicación, si x es cualquier variable que no se use en la demostración, el teorema anterior nos da que

$$\vdash \gamma_1 \wedge \cdots \wedge \gamma_n \to \neg\alpha(x),$$

luego $\Gamma \vdash \neg\alpha(x)$, pero esta deducción se puede completar pasando a $\bigwedge u\, \neg\alpha(u)$ y de ahí a $\neg\bigvee u\, \alpha(u)$, con lo que Γ resulta ser contradictorio, ya que contiene la sentencia $\bigvee u\, \alpha(u)$. ∎

Ahora consideramos un conjunto Γ de sentencias de un lenguaje formal $\mathcal{L}$ y vamos a ver que podemos ejemplificar simultáneamente todas las sentencias de Γ. Para ello llamamos $\mathcal{L}^*$ al lenguaje formal que resulta de añadirle a $\mathcal{L}$ un conjunto infinito C de nuevas constantes.[10]

[10] Esto requiere que haya infinitos números naturales que no sean signos de $\mathcal{L}$, lo cual es una condición que podemos añadir a la definición de lenguaje formal sin pérdida de generalidad, pues podemos elegir los números naturales con los que representamos cada signo de un lenguaje formal, y nada nos impide elegirlos de modo que queden infinitos sin usar.

Sea S el conjunto de todas las semifórmulas de $\mathcal{L}^*$ con una única variable libre (de tipo ligado). El teorema 7.23 nos proporciona una enumeración creciente $F : \mathbb{N} \longrightarrow S$, de modo que podemos representar $F(n) \equiv \alpha_n(u_n)$. Por simplicidad escribiremos $\alpha_n(u)$, entendiendo que u es una variable distinta en cada semifórmula α_n. También podemos definir recurrentemente c_n como la mínima constante de C que no aparece en ninguna de las semifórmulas $\alpha_0, \ldots, \alpha_n$ y es mayor que todas las constantes c_i, con $i < n$. Definimos entonces las sentencias

$$\eta_n \equiv \bigvee u\, \alpha_n(u) \to \alpha_n(c_n),$$

de $\mathcal{L}^*$, de modo que cada c_n no aparece en las sentencias η_i con $i < n$, y es fácil ver que el principio de Δ^0_1-comprensión nos permite definir[11] el conjunto E formado por todas ellas.

Llamamos $\Gamma^+ = \Gamma \cup E$. El teorema anterior nos permite probar que Γ^+ es consistente. En efecto, si fuera contradictorio, existiría una deducción de una contradicción con premisas en Γ^+, pero en ella sólo aparecería un número finito de premisas de E, que estarían entre $\eta_0, \ldots, \eta_{n-1}$, para cierto n. Así tendríamos que $\Gamma_n = \Gamma \cup \{\eta_0, \ldots, \eta_{n-1}\}$ sería contradictorio. Sin embargo, por Σ^0_1-inducción sobre n podemos probar que estos conjuntos son consistentes.

En efecto, estamos suponiendo que $\Gamma_0 = \Gamma$ es consistente y, si Γ_n es consistente pero no lo fuera Γ_{n+1}, entonces $\Gamma_n \vdash \neg\eta_n$ (por reducción al absurdo), luego $\Gamma_n \vdash \bigvee u\, \alpha_n(u) \wedge \neg\alpha_n(c_n)$, pero entonces $\Gamma_n \cup \{\bigvee u\, \alpha_n(u)\}$ es consistente (ya que si pudiéramos probar una contradicción con estas premisas, también podríamos probarla a partir de Γ_n, pues la última premisa se puede deducir), y el teorema anterior nos da (teniendo en cuenta que c_n no aparece en $\Gamma_n \cup \{\bigvee u\, \alpha_n(u)\}$) que $\Gamma_n \cup \{\bigvee u\, \alpha_n(u), \alpha_n(c_n)\}$ es consistente, cuando tenemos que de él se deducen tanto $\alpha_n(c_n)$ como su negación.

Con esto tenemos todo lo necesario para demostrar un teorema fundamental:

Teorema 7.37 ($\mathbf{LKD_0}$) (Teorema de completitud semántica de Gödel) *Un conjunto de fórmulas de un lenguaje formal es consistente si y sólo si tiene un modelo.*

En realidad demostraremos este teorema sin usar directamente el lema de König débil. Usaremos únicamente el lema de Lindenbaum. Luego demostraremos que el teorema de completitud implica LKD, con lo que las tres afirmaciones serán, de hecho, equivalentes sobre ACR_0.

Continuando con el razonamiento precedente, a partir de un conjunto consistente Γ de sentencias de $\mathcal{L}$ hemos construido el conjunto Γ^+ de sentencias de $\mathcal{L}^*$ y ahora el lema de Lindenbaum nos da que existe un conjunto consistente maximal Γ^* de sentencias de $\mathcal{L}^*$ que contiene a Γ^+. Observemos que Γ^* está *ejemplificado* en el sentido siguiente:

[11] La definición de E es totalmente constructiva, de modo que el conjunto E puede definirse mediante una fórmula en ARP exactamente igual que se definen los términos, las fórmulas, las deducciones, etc.

Una sentencia de la forma $\bigvee u\,\alpha(u)$ *de* $\mathcal{L}^*$ *está en* Γ^* *si y sólo si existe una constante* $c \in C$ *tal que* $\alpha(c) \in \Gamma^*$.

En efecto, existe un n tal que la sentencia dada es $\bigvee u\,\alpha_n(u)$, y entonces, llamando c a la constante c_n, tenemos que $\eta_n \equiv \bigvee u\,\alpha(u) \to \alpha(c)$ está en Γ^+, luego en Γ^*. Por lo tanto, si $\bigvee u\,\alpha(u) \in \Gamma^*$, se cumple que $\Gamma^* \vdash \alpha(c)$ y, por la maximalidad, $\alpha(c) \in \Gamma^*$.

Recíprocamente, si $\alpha(c) \in \Gamma^*$, obviamente $\Gamma^* \vdash \bigvee u\,\alpha(u)$, luego $\bigvee u\,\alpha(u) \in \Gamma^*$ por la consistencia maximal.

A su vez, esto implica un resultado análogo para generalizaciones:

Una sentencia $\bigwedge u\,\alpha(u)$ *de* $\mathcal{L}^*$ *está en* Γ^* *si y sólo si para toda constante* $c \in C$ *se cumple* $\alpha(c) \in \Gamma^*$.

Si $\bigwedge u\,\alpha(u) \in \Gamma^*$ pero existiera una constante $c \in C$ tal que $\alpha(c) \notin \Gamma^*$, entonces $\neg\alpha(c) \in \Gamma^*$, luego $\Gamma^* \vdash \neg\bigwedge u\,\alpha(u)$, y así Γ^* sería contradictorio.

Recíprocamente, si $\bigwedge u\,\alpha(u) \notin \Gamma^*$, entonces $\neg\bigwedge u\,\alpha(u) \in \Gamma^*$, luego $\Gamma^* \vdash \bigvee u\,\neg\alpha(u)$, luego por la propiedad precedente existe una constante $c \in C$ tal que $\neg\alpha(c) \in \Gamma^*$, luego $\alpha(c) \notin \Gamma^*$.

Vamos a construir un modelo de Γ de universo[12]

$$M \equiv \{c \mid c \in C \wedge \bigwedge c' < c\, \ulcorner c' = c \urcorner \notin \Gamma^*\}.$$

En primer lugar observamos lo siguiente:

Si t *es un designador de* $\mathcal{L}^*$, *existe una única constante* $c \in M$ *tal que* $\ulcorner c = t \urcorner \in \Gamma^*$.

En efecto, se cumple que $\vdash \bigvee u(u = t)$, luego $\bigvee u\,(u = t) \in \Gamma^*$, luego hemos visto que existe $c \in C$ tal que $\ulcorner c = t \urcorner \in \Gamma^*$, y podemos tomar la mínima constante c que cumple esto. Esto hace que $c \in M$, pues si existe otra constante $c' < c$ tal que $\ulcorner c' = c \urcorner \in \Gamma^*$, entonces $\Gamma^* \vdash \ulcorner c' = t \urcorner$, lo cual contradice la minimalidad de c.

Similarmente concluimos que c es única, pues si hubiera otra c' (por ejemplo, con $c' < c$), tendríamos que $\ulcorner c = t \urcorner, \ulcorner c' = t \urcorner \in \Gamma^*$, luego $\Gamma^* \vdash c' = c$, luego $\ulcorner c' = c \urcorner \in \Gamma^*$, en contra de la definición de M.

Así pues, podemos definir

$$I \equiv \{u \mid \bigvee tc \le u(u = \langle t, c\rangle_2 \wedge t \in \mathrm{Des}(\mathcal{L}^*) \wedge c \in M \wedge \ulcorner c = t \urcorner \in \Gamma^*)\},$$

con lo que $I : \mathrm{Des}(\mathcal{L}^*) \longrightarrow M$ es la aplicación que a cada designador t de $\mathcal{L}^*$ le asigna la única constante de M que cumple $(I(t) = t) \in \Gamma^*$.

Más en general, se cumple:

[12]Usamos ángulos de Quine para representar las fórmulas de $\mathcal{L}$ o $\mathcal{L}^*$ únicamente cuando sin ellos las expresiones puedan resultar ambiguas o simplemente difíciles de leer.

Si t_1, t_2 son designadores de $\mathcal{L}^*$, entonces $I(t_1) = I(t_2)$ si y sólo si $\ulcorner t_1 = t_2 \urcorner \in \Gamma^*$.

En efecto, si $I(t_1) = I(t_2) = c$, entonces $\ulcorner c = t_1 \urcorner, \ulcorner c = t_2 \urcorner \in \Gamma^*$, luego tenemos que $\Gamma^* \vdash t_1 = t_2$, luego $\ulcorner t_1 = t_2 \urcorner \in \Gamma^*$. Recíprocamente, si $\ulcorner t_1 = t_2 \urcorner \in \Gamma^*$ y $c = I(t_1)$, también $\ulcorner c = t_1 \urcorner \in \Gamma^*$, luego $\Gamma^* \vdash c = t_2$, luego $I(t_2) = c$.

Ahora podemos restringir a M los resultados precedentes y añadir uno más sobre existencia con unicidad:

1. *Una sentencia de la forma $\bigvee u\, \alpha(u)$ de $\mathcal{L}^*$ está en Γ^* si y sólo si existe una constante $c \in M$ tal que $\alpha(c) \in \Gamma^*$.*
2. *Una sentencia $\bigwedge u\, \alpha(u)$ de $\mathcal{L}^*$ está en Γ^* si y sólo si para toda constante $c \in M$ se cumple $\alpha(c) \in \Gamma^*$.*
3. *Una sentencia $\overset{1}{\bigvee} u\, \alpha(u)$ de $\mathcal{L}^*$ está en Γ^* si y sólo si existe una única constante $c \in M$ tal que $\alpha(c) \in \Gamma^*$, y en tal caso $I(u|\alpha(u)) = c$.*

En efecto: 1. Si $\bigvee u\, \alpha(u) \in \Gamma^*$, hemos probado que existe una constante $c' \in C$ tal que $\alpha(c') \in \Gamma^*$, así como que existe una constante $c \in M$ tal que $\ulcorner c = c' \urcorner \in \Gamma^*$. Entonces $\Gamma^* \vdash \alpha(c)$, luego $\alpha(c) \in \Gamma^*$. El recíproco es trivial.

2. Si para toda constante $c \in M$ se cumple $\alpha(c) \in \Gamma^*$, para toda constante $c' \in C$ se cumple que existe $c \in M$ tal que $\ulcorner c = c' \urcorner \in \Gamma^*$, con lo que $\Gamma^* \vdash \alpha(c')$, luego $\alpha(c') \in \Gamma^*$ y hemos visto que esto implica que $\bigwedge u\, \alpha(u) \in \Gamma^*$.

3. Por definición, $\overset{1}{\bigvee} u\, \alpha(u) \equiv \bigvee v \bigwedge u(\alpha(u) \leftrightarrow u = v)$. Por 1. esta sentencia está en Γ^* si y sólo si existe $c \in M$ tal que $\bigwedge u(\alpha(u) \leftrightarrow u = c) \in \Gamma^*$ y por 2. esto sucede si y sólo si para toda constante $c' \in M$ se cumple $(\alpha(c') \leftrightarrow c' = c) \in \Gamma^*$, lo cual equivale a su vez a que $\alpha(c') \in \Gamma^*$ si y sólo si $c' = c$. Así acabamos de probar que $\overset{1}{\bigvee} u\, \alpha(u) \in \Gamma^*$ si y sólo si existe una constante $c \in M$ que es la única para la que se cumple $\alpha(c) \in \Gamma^*$.

Además, si $c' = I(u|\alpha(u))$, tenemos que $(c' = u|\alpha(u)) \in \Gamma^*$, luego si suponemos que $\overset{1}{\bigvee} u\, \alpha(u) \in \Gamma^*$ tenemos $\Gamma^* \vdash \alpha(c')$, luego $\alpha(c') \in \Gamma$, luego $c' = c$, luego $I(u|\alpha(u)) = c$.

Por otra parte, como $\vdash (u|u = u) = (v|v = v)$, tenemos que $d = I(u|u = u)$ no depende de la variable u, por lo que nos referiremos a esta constante $d \in M$ como "la descripción impropia".

Ahora es fácil dotar a M de estructura de premodelo de $\mathcal{L}^*$. Por razones técnicas, llamamos $\bar{M}$ al conjunto que hasta ahora habíamos llamado M, y definimos

$$M^0 \equiv \{v \mid \bigvee c \le v(v = \langle 0, c\rangle_2 \wedge c \in \bar{M})\},$$

de modo que $M^0_0 = \bar{M}$. A su vez, definimos

$$M^1 \equiv \{v \mid \bigvee cc' \le v(v = \langle 1, \langle c, c'\rangle_2\rangle_2 \wedge c \in \mathrm{Const}(\mathcal{L}^*) \wedge c' = I(c))\}$$
$$\cup \{\langle 1, \langle |, d\rangle_2\rangle_2\}.$$

Así

$$M_1^1 : \mathrm{Const}(\mathcal{L}^*) \cup \{|\} \longrightarrow \bar{M}$$

es la aplicación dada por $M_1^1(c) = I(c)$ y $M_1^1(|) = d$.

Ahora tenemos que definir M^2 de modo que, para cada funtor n-ádico f de $\mathcal{L}$, se cumpla que $(M_2^2)_f : \bar{M}^n \longrightarrow \bar{M}$ y para cada relator n-ádico R de $\mathcal{L}$ se cumpla que $(M_2^2)_R \subset \bar{M}^n$. Escribir explícitamente la definición de M^2 sería farragoso, así que vamos a indicar únicamente cómo se definen las funciones $(M_2^2)_f$ y relaciones $(M_2^2)_R$:

$$(M_2^2)_f(c_1, \ldots, c_n) = I(f(c_1, \ldots, c_n)),$$

$$(M_2^2)_R(c_1, \ldots, c_n) \leftrightarrow R(c_1, \ldots, c_n) \in \Gamma^*.$$

(Notemos que así $(M_2^2)_=$ es la identidad en $\bar{M}$). Es pura rutina comprobar que estas definiciones pueden reunirse en una única definición de un conjunto M^2. Así, si llamamos $M = M^0 \cup M^1 \cup M^2$, tenemos que M es un premodelo de $\mathcal{L}^*$. Pero en realidad vamos a completar la definición de modo que $M = M^0 \cup M^1 \cup M^2 \cup M^3$ sea, de hecho, un modelo. Definir M^3 equivale a definir $M(t)[v]$ y $M \vDash \alpha[v]$ para todos los pares de P_M.

Si $\langle \theta, v \rangle_2 \in P_M$, de modo que $x_1, \ldots, x_n$ son las variables (libres o ligadas) que están libres en la semiexpresión θ, entonces cada $v(x_i)$ es una constante de $\mathcal{L}^*$, luego podemos definir

$$v[\theta] = \mathtt{S}^{v(x_1)\cdots v(x_n)}_{x_1 \cdots x_n}\theta,$$

y así $v[\theta]$ es un designador o una sentencia de $\mathcal{L}^*$ según si θ es un semitérmino o una semifórmula de $\mathcal{L}^*$. En particular, si θ no tiene variables libres, tenemos que $v[\theta] = \theta$. Definimos

$$M(t)[v] = I(v[t]), \qquad M \vDash \alpha[v] \leftrightarrow v[\alpha] \in \Gamma^*.$$

Es claro que podemos definir M^3 adecuadamente para que la función $M(t)[v]$ y la relación $M \vDash \alpha[v]$ sean las que acabamos de definir. Ahora vamos a comprobar que, con ellas, M es ciertamente un modelo de $\mathcal{L}^*$. En efecto:

1. $M(x)[v] = I(v[x]) = I(v(x)) = v(x)$.
2. $M(c)[v] = I(v[c]) = I(c) = M(c)$.
3. $M(f(t_1, \ldots, t_n))[v] = I(v[f(t_1, \ldots, t_n)]) = I(f(v[t_1], \ldots, v[t_n]))$.

   En el último paso hemos usado la definición de la sustitución de variables por términos. Llamemos $c_i = M(t_i)[v] = I(v[t_i])$. Por definición de I, se cumple que $(c_i = v[t_i]) \in \Gamma^*$, luego

   $$\Gamma^* \vdash f(v[t_1], \ldots, f[t_n]) = f(c_1, \ldots, c_n),$$

   luego la igualdad está en Γ^*, luego

   $$I(f(v[t_1], \ldots, v[t_n])) = I(f(c_1, \ldots, c_n)) = M(f)(c_1, \ldots, c_n) =$$

   $$M(f)(M(t_1)[v], \ldots, M(t_n)[v]).$$

4. $M \vDash R(t_1,\dots,t_n)[v] \leftrightarrow v[R(t_1,\dots,t_n)] \in \Gamma^* \leftrightarrow R(v[t_1],\dots v[t_n]) \in \Gamma^*$.
   Con la misma notación del apartado precedente, como $(c_i = v[t_i]) \in \Gamma^*$, tenemos que
   $$(R(v[t_1],\dots v[t_n]) \leftrightarrow R(c_1,\dots,c_n)) \in \Gamma^*,$$
   luego
   $$R(v[t_1],\dots v[t_n]) \in \Gamma^* \leftrightarrow R(c_1,\dots,c_n) \in \Gamma^* \leftrightarrow M(R)(c_1,\dots,c_n) \leftrightarrow$$
   $$M(R)(M(t_1)[v],\dots,M(t_n)[v]).$$
5. $M \vDash \neg\alpha[v] \leftrightarrow \neg v[\alpha] \in \Gamma^* \leftrightarrow v[\alpha] \notin \Gamma^* \leftrightarrow \neg M \vDash \alpha[v]$.
   $$M \vDash (\alpha \vee \beta)[v] \leftrightarrow v[\alpha] \vee v[\beta] \in \Gamma^* \leftrightarrow v[\alpha] \in \Gamma^* \vee v[\beta] \in \Gamma^*$$
   $$\leftrightarrow M \vDash \alpha[v] \vee M \vDash \beta[v].$$
6. Si $\alpha(u)$ es una semifórmula, entonces, si llamamos $v^*[\alpha]$ a la fórmula que resulta de sustituir las variables libres x_i de α por $v(x_i)$ pero sin sustituir la variable ligada u, aplicando en el primer paso la definición de sustitución, tenemos que
   $$M \vDash \bigwedge u\, \alpha(u)[v] \leftrightarrow \bigwedge u\, v^*[\alpha](u) \in \Gamma^* \leftrightarrow \bigwedge c \in M\ v_u^c[\alpha] \in \Gamma^*$$
   $$\leftrightarrow \bigwedge c \in M \vDash \alpha[v_u^c],$$
   y
   $$M \vDash \bigvee u\, \alpha(u)[v] \leftrightarrow \bigvee u\, v^*[\alpha](u) \in \Gamma^* \leftrightarrow \bigvee c \in M\ v_u^c[\alpha] \in \Gamma^*$$
   $$\leftrightarrow \bigvee c \in M \vDash \alpha[v_u^c].$$
7. Con la notación precedente, $M(u|\alpha(u))[v] = I(u|v^*[\alpha](u))$. Supongamos que existe un único $c \in M$ tal que $M \vDash \alpha(u)[v_u^c]$, que es lo mismo que $M \vDash v^*[\alpha](c)$. Hemos probado que esto implica que $I(u|v^*[\alpha](u)) = c$.
   Si, por el contrario, no existe un único $c \in M$ tal que $M \vDash \alpha(u)[v_u^c]$, es decir, tal que $M \vDash v^*[\alpha][c]$, esto implica que $\overset{1}{\bigvee} u\, v^*[\alpha](c) \notin \Gamma^*$, luego $\neg\overset{1}{\bigvee} u\, v^*[\alpha](c) \in \Gamma^*$, luego $\Gamma^* \vdash u|v^*[\alpha](u) = u|u = u$, luego $I(u|v^*[\alpha](u)) = I(u|u = u)$, que es la descripción impropia de M.

Con esto tenemos probado que M es un modelo de $\mathcal{L}^*$ y, además, para cada sentencia α de $\mathcal{L}^*$, tenemos que $M \vDash \alpha$ si y sólo si $\alpha \in \Gamma^*$. En otras palabras, $\Gamma^* = T(M^*)$ es el conjunto de todas las sentencias verdaderas en M. En particular, $M \vDash \Gamma$.

Obviamente, podemos modificar la definición de M para que no interprete las constantes de C, ni las fórmulas de $\mathcal{L}^*$ que las contengan, y así M se restringe a un modelo de $\mathcal{L}$ tal que $M \vDash \Gamma$.

Así hemos probado que todo conjunto consistente de sentencias de $\mathcal{L}$ tiene un modelo. Falta probar que lo mismo vale para conjuntos de fórmulas arbitrarias, pero esto es inmediato sin más que tener en cuenta que, si $\alpha(x_1, \ldots, x_n)$ es una fórmula con las variables libres indicadas y definimos su clausura universal como $\alpha^c \equiv \bigwedge u_1 \cdots u_n \alpha(u_1, \ldots, u_n)$, cualquier modelo cumple

$$M \vDash \alpha \text{ si y sólo si } M \vDash \alpha^c,$$

luego, si llamamos Γ^c al conjunto de todas las clausuras universales de las fórmulas de Γ, se cumple que $M \vDash \Gamma$ si y sólo si $M \vDash \Gamma^c$. Por lo tanto, si Γ es un conjunto consistente de fórmulas de $\mathcal{L}$, vemos que Γ^c es un conjunto consistente de sentencias de $\mathcal{L}$ (pues si de Γ^c se sigue una contradicción, también podemos obtenerla a partir de Γ, demostrando cada premisa α^c a partir de α usando IG). Por la parte ya probada, Γ^c tiene un modelo M que, según acabamos de observar, es también un modelo de Γ.

Ya hemos señalado tras la definición de consistencia que el recíproco es evidente, es decir, que si Γ tiene un modelo entonces es consistente, por lo que tenemos probado el teorema de completitud. ∎

Aunque el teorema al que hemos llamado "teorema de completitud semántica" es el más usado en la práctica, el nombre es más adecuado para la consecuencia siguiente:

Teorema 7.38 ($\mathbf{LKD_0}$) (Teorema de completitud semántica de Gödel) *Si Γ es un conjunto de fórmulas de un lenguaje formal y α es una fórmula tal que $\Gamma \vDash \alpha$, es decir, que α es verdadera en todos los modelos de Γ, entonces $\Gamma \vdash \alpha$.*

Demostración: Si no $\Gamma \vdash \alpha$, entonces $\Gamma \cup \{\neg\alpha\}$ es consistente, pues si fuera contradictorio podríamos deducir α de Γ por reducción al absurdo. Por la versión que hemos probado del teorema de completitud, $\Gamma \cup \{\neg\alpha\}$ tiene un modelo M, que es, pues, un modelo de Γ en el que α es falsa. ∎

Esto significa que el cálculo deductivo es completo en el sentido de que si no permite deducir una fórmula α de un conjunto de premisas Γ, ello se debe necesariamente a que es posible que las premisas sean verdaderas en un modelo en el cual α sea falsa, por lo que no podemos decir que α sea consecuencia lógica de las premisas. Dicho al revés, siempre que una fórmula es una consecuencia lógica de unas premisas en el sentido semántico de que es verdadera siempre que las premisas son verdaderas, el cálculo deductivo que hemos definido permite deducir formalmente α a partir de las premisas.

Ahora ya podemos probar (en ACR_0):

Teorema 7.39 *Las afirmaciones siguientes son equivalentes:*

1. **El lema de König débil:** *Todo árbol binario infinito tiene un camino.*
2. **El lema de Lindenbaum:** *Todo conjunto consistente de sentencias de un lenguaje formal de primer orden $\mathcal{L}$ puede extenderse a un conjunto consistente maximal.*

3. **El teorema de completitud semántica** *Un conjunto de fórmulas de un lenguaje formal de primer orden es consistente si y sólo si tiene un modelo.*

4. **El teorema de compacidad** *Si* Γ *es un conjunto de fórmulas de un lenguaje formal de primer orden tal que cada subconjunto finito de* Γ *tiene un modelo, entonces* Γ *tiene un modelo.*

DEMOSTRACIÓN: Ya hemos demostrado que $1 \Rightarrow 2 \Rightarrow 3$. La implicación $3 \Rightarrow 4$ es muy simple: si Γ no tiene un modelo es que no es consistente, luego existe una deducción de una contradicción con premisas en Γ, pero en ella sólo se usará un conjunto finito de premisas Γ_0, que será, pues, contradictorio, luego no tendrá un modelo.

Falta probar que $4 \Rightarrow 1$. Para ello consideramos un lenguaje formal de primer orden cuyos únicos signos eventuales sean un relator monádico R y un conjunto infinito de constantes $c_0, c_1, c_2, \ldots$ Llamaremos $p_n^0 \equiv R(c_n)$ y $p_n^1 \equiv \neg R(c_n)$. Formalmente, tenemos una aplicación $P : \mathbb{N} \times \{0,1\} \longrightarrow \mathrm{Sent}(\mathcal{L})$ tal que $P(n,i) = p_n^i$.

Supongamos ahora que $A \subset 2^{<\omega}$ es un árbol infinito. Para cada n, llamamos $A_n \equiv \{s \mid s \in A \wedge \ell(s) = n\}$, que es un conjunto finito no vacío. Para cada $s \in A$ de altura $n \geq 1$, llamamos $\sigma(s) = \bigwedge_{i<n} p_i^{s_i}$. Para cada $n \geq 1$, definimos

$$\sigma_n \equiv \bigvee_{s \in A_n} \sigma(s).$$

Notemos que podemos definir el conjunto Γ formado por todas las sentencias σ_n. Por ejemplo, teniendo en cuenta que cada una es una disyunción de sentencias de longitud mayor o igual que n, se cumple que $n \leq \sigma_n$, por lo que podemos definir

$$\Gamma \equiv \{\alpha \mid \bigvee n \leq \alpha\, \alpha = \sigma_n\},$$

y la definición es Δ_1^0. También podemos definir

$$\Gamma_n \equiv \{\alpha \mid \bigvee i \leq n\, \alpha = \sigma_i\}.$$

Observemos que, para cada $s \in 2^{<\omega}$, se cumple que $\sigma(s)$ tiene un modelo M. Si $\ell(s) = n > 0$, basta tomar como universo el conjunto $M_n = I_n$, interpretar

$$M_n(c_i) = \begin{cases} i & \text{si } i < n, \\ 0 & \text{si } i \geq n. \end{cases}$$

A su vez, definimos $M_n(R)(i) \leftrightarrow s_i = 1$. El hecho de que el universo de M_n sea finito permite convertir sin dificultad al premodelo que acabamos de definir en un modelo de $\mathcal{L}$ de modo que $M_n \vDash \sigma(s)$. A su vez, esto implica trivialmente que $M_n \vDash \sigma_n$.

Por otra parte, si $\ell(s) = n+1$, tenemos que $\sigma(s) = \sigma(s|_n) \wedge p_n^{s_n}$, por lo que $\vdash \sigma(s) \to \sigma(s|_n)$, y esto hace que $\vdash \sigma_{n+1} \to \sigma_n$, pues σ_{n+1} y σ_n son dos

disyunciones tales que cada término de la primera implica uno de los términos de la segunda. Por lo tanto, $M_n \vDash \Gamma_n$, y esto implica que todos los subconjuntos finitos de Γ tienen un modelo. Por el teorema de compacidad, podemos concluir que Γ tiene un modelo M.

Definimos $F : \mathbb{N} \longrightarrow \{0,1\}$ mediante $F(n) = 1 \leftrightarrow M \vDash p_n$ y basta ver que F es un camino en A. Para ello tenemos que probar que $\bigwedge n\, F^n \in A$. Tomemos un n y supongamos que $t = F^n \notin A$. Equivalentemente, $t \notin A_n$. Entonces, si $s \in A_n$, existe un $i < n$ tal que $s_i \neq t_i$. Si $t_i = 1$ tenemos que $M \vDash p_i$, mientras que $s_i = 0$, luego $\vdash \sigma(s) \to \neg p_i$, por lo que $M \vDash \neg\sigma(s)$. Igualmente, si $t_i = 0$ tenemos que $M \vDash \neg p_i$, mientras que $\vdash \sigma(s) \to p_i$, luego $M \vDash \neg\sigma(s)$ en cualquier caso, y esto vale para todo $s \in A_n$, luego $M \vDash \neg\sigma_n$, y tenemos una contradicción. ∎

7.5 Modelos no estándar

Según acabamos de ver, el teorema de completitud semántica nos asegura que el cálculo deductivo es exactamente lo que tiene que ser, en el sentido de que no sería admisible otro distinto de $K_{\mathcal{L}}$ que permitiera extraer más consecuencias de unas premisas dadas, y es el ejemplo básico de que definir un cálculo deductivo formal no es meramente dar arbitrariamente unos axiomas y reglas de inferencia y ya está, sino que es necesario (o, al menos, deseable y —dado que es posible— exigible) comprobar que tales axiomas y reglas realmente capturan la noción informal de deducción lógica, precisamente en el sentido en que lo justifica el teorema de completitud semántica, lo cual requiere algo de metamatemática adicional, y ya hemos comprobado que los supuestos metamatemáticos necesarios para probar el teorema de completitud semántica son (a lo sumo) los que expresa la teoría LKD_0.

Ahora vamos a ver que, sorprendentemente, el teorema de completitud semántica también pone de manifiesto una debilidad esencial de la lógica formal respecto del razonamiento intuitivo. Vamos a ver que ninguna teoría axiomática formal puede determinar el concepto de "número natural" o el de "conjunto finito".

En efecto, supongamos que el lector pretende determinar el concepto de "número natural" a partir de una definición formal adecuada en una teoría axiomática formal adecuada. Dejamos que el lector elija la teoría T que considere oportuna, sobre el lenguaje formal $\mathcal{L}$ que considere oportuno. En dicha teoría, el lector puede elegir una fórmula, que podemos representar por $x \in \mathbb{N}$, que constituya su definición formal de número natural.[13] En la teoría T se tendrán que probar algunos hechos básicos sobre los números naturales. Como mínimo, tendrá que haber un designador de $\mathcal{L}$ al que podemos llamar 0 y un término x' de $\mathcal{L}$ con la variable x libre de modo que en T se puedan demostrar los teoremas siguientes:

[13] Por ejemplo, un intento puede ser la fórmula $x \in \omega$ definida en 5.14 en la teoría básica de conjuntos B.

1. $0 \in \mathbb{N}$,
2. $\bigwedge u(u \in \mathbb{N} \to u' \in \mathbb{N})$,
3. $\bigwedge u(u \in \mathbb{N} \to u' \neq 0)$,
4. $\bigwedge uv(u \in \mathbb{N} \wedge v \in \mathbb{N} \wedge u' = v' \to u = v)$.

No vamos a suponer nada más. El lector puede diseñar una teoría T adecuada y, en ella, formular una definición adecuada del concepto de "número natural" de modo que sea posible demostrar todos los resultados razonables sobre números naturales que considere necesarios para asegurar que los objetos que cumplen su definición serán necesariamente los números naturales en el sentido intuitivo que sabemos darle a este concepto. La única limitación que imponemos a la creatividad del lector es que la teoría T resultante tiene que ser consistente, pues si fuera contradictoria, en ella se podría demostrar cualquier cosa y no tendría ningún valor.

Bajo estos mínimos supuestos podemos definir numerales en $\mathcal{L}$, es decir, los términos de la forma

$$0, \quad 0', \quad 0'', \quad 0''', \quad 0'''', \quad \dots$$

Más precisamente, a cada número natural informal n le podemos asignar el numeral que consta de un 0 y n "comitas" o, en otros términos:

$$0^{(0)} \equiv 0, \qquad 0^{(n+1)} \equiv (0^{(n)})'.$$

Aplicando repetidamente los teoremas 1 y 2 vemos que, para todo número natural n, se cumple $\underset{T}{\vdash} 0^{(n)} \in \mathbb{N}$.

Si la teoría T es consistente, sabemos que tendrá un modelo M, y si el lector ha tenido éxito caracterizando formalmente los números naturales, debería cumplirse que los únicos objetos $a \in M$ que cumplen $M \vDash (x \in \mathbb{N})[v^a_x]$ fueran los objetos denotados por los numerales, es decir:

$$a_0 = M(0), \quad a_1 = M(0'), \quad a_2 = M(0''), \quad a_3 = M(0'''), \quad \dots$$

El problema es qué requisitos imponer a una definición formal de "número natural" que nos permitan asegurar que, en cualquier modelo de nuestra teoría, los objetos que cumplen la definición sean el cero y los que se obtienen del cero mediante un número finito de aplicaciones de la operación "siguiente". ¡Pero vamos a ver que eso es imposible!

Tomemos un lenguaje $\mathcal{L}^*$ que conste de los mismos signos que $\mathcal{L}$ más una nueva constante c y consideremos las fórmulas

$$\phi_n(x) \equiv x \in \mathbb{N} \wedge x \neq 0^{(0)} \wedge x \neq 0^{(1)} \wedge \cdots \wedge x \neq 0^{(n)}.$$

Usando las cuatro sentencias que suponemos demostrables en T es fácil ver que

$$\underset{T}{\vdash} \phi_n(0^{(n+1)}).$$

Veamos ahora que, en la teoría sobre $\mathcal{L}^*$ que tiene los mismos axiomas que T, no se puede demostrar la fórmula $\neg\phi_n(c)$. En efecto, si se pudiera, llamamos α a la conjunción de los axiomas de T que aparecen en la demostración. Podemos suponer que son sentencias, sin más que cuantificar universalmente todas sus variables libres. Entonces el teorema de deducción nos da que

$$\vdash (\alpha \to \neg\phi_n(c)),$$

y el teorema 7.35 implica entonces que $\vdash (\alpha \to \neg\phi_n(x))$ (y la demostración muestra que en la demostración no aparece la variable c, por lo que se trata de una demostración en la teoría T original, sobre el lenguaje $\mathcal{L}$), lo que a su vez implica que $\underset{T}{\vdash} \neg\phi_n(x)$, de donde, introduciendo el generalizador y eliminándolo después, concluimos que $\underset{T}{\vdash} \neg\phi_n(0^{(n+1)})$, y resulta que la teoría T es contradictoria, en contra de lo supuesto.

Recíprocamente, si T es consistente, tenemos que $\neg\phi_n(c)$ no es un teorema de T, luego la teoría T_n que resulta de tomar como axioma adicional la fórmula $\phi_n(c)$ también es consistente. (Si fuera contradictoria, suponiendo $\phi_n(c)$ en T podríamos demostrar una contradicción, luego tendríamos una demostración de $\neg\phi_n(c)$ por reducción al absurdo).

Llamamos Γ al conjunto formado por los axiomas de T más todas las fórmulas $\phi_n(c)$, y observamos que todo subconjunto finito de Γ es consistente.

En efecto, Γ_0 es un subconjunto finito de Γ y fuera contradictorio, lo seguiría siendo al añadirle más fórmulas, luego podemos suponer que consta de un número finito de axiomas de T más las fórmulas $\phi_0(c), \ldots, \phi_n(c)$. Ahora bien, si $i < n$, tenemos que $\phi_n(c)$ implica $\phi_i(c)$, luego podemos eliminar todas las $\phi_i(c)$ con $i < n$ y concluimos que también sería contradictoria la teoría T_n, cuando hemos visto que no es así.

Por el teorema de compacidad 7.39, tenemos que Γ es consistente, es decir, que la teoría T^* que resulta de añadirle a T todos los axiomas $\phi_n(c)$ es consistente, y por el teorema de completitud tiene un modelo M, que, en particular, es un modelo de T.

Llamemos

$$a_0 = M(0), \quad a_1 = M(0'), \quad a_2 = M(0''), \quad a_3 = M(0'''), \quad \ldots$$

Puesto que $\underset{T}{\vdash} 0^{(n)} \in \mathbb{N}$ tenemos que todos los a_n son objetos del universo del modelo que cumplen la definición de número natural (cualquiera que sea la que hayamos elegido), pero, si llamamos $\xi = M(c)$, el hecho de que $\underset{T^*}{\vdash} \phi_n(c)$ implica que ξ también cumple la definición de número natural, así como que $\xi \neq a_n$. Por lo tanto, ξ es un objeto del universo del modelo M que cumple la definición formal de número natural que hemos adoptado (cualquiera que sea), pero que es distinto de todos los números denotados por los numerales.

En suma: hemos probado que existe un modelo M de la teoría T en la que hay un objeto que cumple nuestra definición de número natural, pero es distinto

del cero (el objeto denotado por 0), del uno (el objeto denotado por $0'$), del dos (el objeto denotado por $0''$) y, en general, que es distinto de todos los números naturales que se obtienen del cero aplicando la operación "siguiente". Y lo notable es que esto es así sea cual sea la definición que demos de número natural y de lo potente que sea la teoría axiomática en la que formalicemos la definición. Aunque tratemos de dar una definición formal de "número natural" en una teoría de conjuntos potente como ZFC y por más precisiones que incluyamos en ella (mientras no perdamos la consistencia) no podremos evitar que nuestra teoría tenga un modelo en el que nuestra definición la satisfaga un objeto que no sea ninguno de los que se obtienen del objeto que cumple nuestra definición de "cero" aplicándole sucesivamente nuestra definición de "siguiente", con lo que podemos afirmar que cualquier definición formal de "número natural" no determina los números naturales.

El formalista radical que tacha la matemática intuitiva de "imprecisa" tiene que afrontar que la única definición precisa del concepto de "número natural" es la definición intuitiva según la cual los números naturales son conceptos abstractos determinados por el hecho de que hay un primer número natural llamado "cero", por que todo número natural tiene un "siguiente" (distinto de todos los precedentes) y por el hecho de que no hay más números naturales que los que se obtienen del cero mediante sucesivos pasos al siguiente. Esto último no puede ser exigido por ninguna definición formal, luego toda definición formal de número natural es incapaz de precisar lo que la definición intuitiva sí que precisa.

Se dice que modelo M de una teoría T en las condiciones anteriores es un *modelo estándar* si los únicos objetos de su universo que satisfacen la definición de "número natural" son los denotados por los numerales. En caso contrario se trata de un *modelo no estándar,* y los objetos que satisfacen la definición de número natural se dividen en *números naturales estándar* y *números naturales no estándar.* Hay que entender que los números naturales no estándar de un modelo no estándar no son números naturales en el sentido de que no se corresponden con ningún número natural intuitivo, pero son formalmente números naturales en el sentido de que satisfacen la definición formal de número natural que hayamos adoptado.

Hemos demostrado que toda teoría axiomática consistente en la que se haya dado una definición de número natural que cumpla las condiciones mínimas que hemos exigido admite inevitablemente modelos no estándar.

Si en la teoría considerada se puede demostrar al menos el teorema 6.4 (para lo cual basta con que interprete a Q), entonces es obvio que en un modelo no estándar un número natural no estándar es necesariamente mayor que todos los números estándar (lo que dice 6.4 es que todo número menor o igual que un número estándar es estándar). Por ello, a veces a los números naturales no estándar se les llama también "números naturales infinitos", es decir, números naturales (objetos que cumplen formalmente la definición de número natural) que son mayores que todos los números estándar y que, por consiguiente, tienen por debajo infinitos números naturales.

No obstante, hay que insistir en que el hecho de que cualquier teoría aritmética consistente admita modelos no estándar no contradice que cualquiera de las teorías aritméticas que hemos estudiado, como ARP, $\mathrm{I}\Sigma_1$, AP, $\mathrm{KP}_{\mathrm{fin}}$, $\mathrm{ZFC}_{\mathrm{fin}}$, ACR_0, ACA_0, etc., cumplen su cometido de permitirnos demostrar formalmente teoremas sobre números naturales con la garantía de que todos los teoremas se pueden interpretar como afirmaciones verdaderas sobre los números naturales intuitivos,[14] sin perjuicio de que también sean verdaderas sobre otros objetos exóticos (los números naturales de los modelos no estándar de la teoría).

Conjuntos finitos Veamos ahora que sucede lo mismo con el concepto intuitivo de "conjunto finito/infinito": ninguna definición formal determina inequívocamente la finitud.

Aunque se podría probar bajo hipótesis aún más generales, supongamos que el lector fija una teoría axiomática T con las características siguientes:

1. T es consistente.
2. En T hemos definido los números naturales de modo que se cumplan las mismas condiciones mínimas que antes.
3. En T hemos definido una fórmula "x es un conjunto finito", de modo que puede demostrarse la existencia del conjunto I_n de los números naturales menores que uno dado n y que se trata de un conjunto finito.

   En particular, esto se cumple si en T se puede demostrar que el conjunto vacío es finito y que, si x es un conjunto finito, entonces todo conjunto de la forma $x \cup \{a\}$ es finito, así como un principio de inducción aplicable a la fórmula "el conjunto I_n es finito".

En estas condiciones, en un modelo no estándar de T, el conjunto de los números naturales menores que un número no estándar dado es un conjunto que cumple la definición de finitud, pero que contiene a todos los números naturales estándar, por lo que en realidad es infinito.

Así pues, la noción intuitiva de finitud, según la cual un conjunto es finito si es posible enumerar sus elementos de modo que el proceso de enumeración termine, no es formalizable. Cualquier intento razonable de definir formalmente la finitud admitirá una interpretación en la que un conjunto infinito satisfará la definición formal de conjunto finito.

Conjuntos y colecciones de objetos Otra consecuencia de la existencia de modelos no estándar es que no podemos identificar ningún concepto formal de "conjunto" con el de "colección de objetos", en el sentido de que cualquier teoría de conjuntos en la que se puedan definir los números naturales admite modelos en los que existen colecciones de números naturales que no son los elementos de ninguno de los conjuntos de la teoría.

[14] Prescindiendo de tecnicismos, como en que en teorías como ZFC se cumplan tecnicismos como que $3 = \{0, 1, 2\}$, que no es ni verdadero ni falso en términos intuitivos, sino simplemente un sinsentido.

En efecto, en un modelo no estándar M de cualquier teoría de conjuntos, o en cualquier teoría aritmética de segundo orden (y hemos visto que siempre hay modelos no estándar, si la teoría es consistente) no hay ningún objeto (ningún conjunto) al que pertenezcan exactamente los números naturales estándar, supuesto que en la teoría se pueda demostrar que

$$\bigwedge X(X \subset \mathbb{N} \wedge 0 \in X \wedge \bigwedge n(n \in \mathbb{N} \wedge n \in X \to n' \in X) \to X = \mathbb{N}).$$

En efecto, si en M existiera el conjunto de los números naturales estándar, en virtud de este teorema sería el conjunto de todos los números naturales, luego no habría números no estándar. Así, tenemos una colección de objetos bien definida (todos los objetos del modelo M que son denotados por un numeral) que no constituyen ninguno de los conjuntos considerados en la teoría, según la interpretación fijada por M.

En general, cualquier modelo de cualquier teoría de conjuntos razonable permite interpretar los conjuntos de la teoría como ciertas colecciones de objetos, pero no es necesariamente cierto que cualquier colección de objetos del universo del modelo se corresponda necesariamente con un conjunto. Ya conocíamos ejemplos de esta situación, pues sabemos que en un modelo de la teoría de Zermelo Z no puede haber un conjunto que contenga a todos los conjuntos, pero "el conjunto de todos los conjuntos" sería algo muy grande y abstracto. Sin embargo, ahora tenemos ejemplos de meras colecciones de números naturales que no pueden ser conjuntos. Más precisamente ¡la colección de todos los números naturales "de verdad", los estándar, no puede ser un conjunto en un modelo no estándar!

Tenemos así varios ejemplos de reflexiones sobre el alcance de la lógica formal que, si bien no son imprescindibles para usar la lógica como fundamentación de la matemática, son hechos relevantes para entender cabalmente dicho alcance que no puede entender un formalista radical empeñado en que la matemática informal es algo "poco riguroso" que puede ser "orientativo" a efectos pedagógicos, pero totalmente prescindible cuando se trata de ser preciso y riguroso. Por el contrario, acabamos de ver varios razonamientos necesariamente informales totalmente precisos y rigurosos que muestran limitaciones notables muy concretas de la lógica formal que no dejan de ser ciertas porque alguien se niegue a enfrentarse tales razonamientos tachándolos de "informales".

7.6 El axioma de comprensión aritmética

Recordemos (definición 7.5) que la teoría ACA_0 se obtiene de ACR_0 extendiendo el principio de Δ^0_1-comprensión al principio de comprensión aritmética ACA, que afirma que todas las fórmulas aritméticas (las fórmulas de primer orden de $\mathcal{L}^2_a$) definen conjuntos. Esto hace que, del principio de Σ^0_1-inducción de ACR_0, podamos quedarnos únicamente con el caso particular

$$\bigwedge U(0 \in U \wedge \bigwedge u(u \in U \to u' \in U) \to \bigwedge u\, u \in U),$$

pues, en conjunción con el axioma de comprensión aritmética, éste implica por sí solo todos los casos particulares del esquema de inducción para fórmulas aritméticas.

Así pues, ACA_0 consta de un número finito de axiomas más el esquema de comprensión aritmética. Enseguida veremos que, al contrario que AP, la teoría ACA_0 es finitamente axiomatizable, pero antes vamos a ver algunos hechos más elementales. El teorema siguiente lo tenemos ya prácticamente demostrado:

Teorema 7.40 *Las afirmaciones siguientes son equivalentes en* ACR_0*:*

1. *El axioma de comprensión aritmética.*
2. *El principio de* Σ^0_1*-comprensión.*
3. *Si* $F : X \longrightarrow Y$ *y* $A \subset X$*, existe un conjunto* B *tal que*
$$\bigwedge n(n \in B \leftrightarrow \bigvee m\,(m \in A \wedge F(m) = n)).$$
4. *Si* $F : \mathbb{N} \longrightarrow \mathbb{N}$ *es inyectiva, existe un conjunto* R *tal que*
$$\bigwedge n(n \in R \leftrightarrow \bigvee m\,F(m) = n).$$

El conjunto cuya existencia se afirma en 3. no es sino la imagen de A por F, que representaremos por

$$F[A] \equiv B \mid \bigwedge n(n \in B \leftrightarrow \bigvee m(m \in A \wedge F(m) = n)).$$

Demostración: $1 \Rightarrow 2 \Rightarrow 3 \Rightarrow 4$ son triviales, mientras que $2 \Rightarrow 1$ es el teorema 7.9. Sólo falta probar que $4 \Rightarrow 2$, pero esto es consecuencia de 7.24, que afirma que, dada una fórmula de tipo Σ^0_1, o bien define un conjunto finito X, o bien existe una función[15] $F : \mathbb{N} \longrightarrow \mathbb{N}$ inyectiva tal que $F[\mathbb{N}]$ es precisamente el conjunto definido por la fórmula, luego en ambos casos existe el conjunto. ∎

Veamos ahora que el axioma de comprensión aritmética equivale al lema de König que enunciamos a continuación:

Teorema 7.41 *Las afirmaciones siguientes son equivalentes en* ACR_0*:*

1. *El axioma de comprensión aritmética.*
2. *El lema de König: Todo árbol infinito finitamente ramificado tiene un camino.*
3. *Todo árbol infinito en el que cada nodo tiene a lo sumo dos sucesores inmediatos tiene un camino.*

[15] En 7.24 considerábamos funciones $F : \mathbb{N}^1 \longrightarrow \mathbb{N}$, que es más conveniente si vamos a trabajar, en general, con funciones $F : \mathbb{N}^r \longrightarrow \mathbb{N}$, pero es equivalente considerar funciones $F : \mathbb{N} \longrightarrow \mathbb{N}$, pues en ACR_0 podemos definir la biyección natural $\mathbb{N}^1 \longrightarrow \mathbb{N}$.

DEMOSTRACIÓN: $1 \Rightarrow 2$ Sea A un árbol infinito finitamente ramificado. El axioma de comprensión aritmética nos permite definir

$$A^* = \{s \mid s \in A \wedge \bigwedge m \bigvee t(m \leq t \wedge t \in A \wedge s \preceq t)\},$$

es decir, el conjunto de todos los nodos de A que tienen por encima infinitos nodos de A. Como A es infinito, se cumple que $0 \in A^*$ y el hecho de que A sea finitamente ramificado implica que si $s \in A^*$, existe un n tal que $s^\frown\langle n\rangle \in A^*$ (en caso contrario, cada sucesor inmediato de s tendría un número finito de nodos de A por encima, y lo mismo le sucedería a s).

Definimos entonces $H : \mathbb{N}^2 \longrightarrow \mathbb{N}$ mediante

$$H = \{u \mid \bigvee mstn(u = \langle\langle m, s\rangle_2, t\rangle_2 \wedge (s \in A^* \wedge t \in A^* \wedge t = s^\frown\langle n\rangle \wedge$$

$$\bigwedge v < n\, s^\frown\langle v\rangle \notin A^*) \vee (s \notin A^* \wedge t = 0))\}.$$

En otras palabras, $H : \mathbb{N}^2 \longrightarrow \mathbb{N}$ es la función tal que si $s \in A^*$, entonces $H(m, s) = s^\frown\langle n\rangle$, donde n es el mínimo número natural tal que $s^\frown\langle n\rangle \in A^*$, mientras que si $s \notin A^*$, entonces $H(m, s) = 0$. En particular, si $s \in A^*$, tenemos que $s \prec H(m, s)$. La función H nos permite definir por recurrencia (véase la nota tras el teorema 7.16) la función $F : \mathbb{N} \longrightarrow \mathbb{N}$ dada por

$$F(0) = 0, \qquad F(n + 1) = H(n, F(n)),$$

y por inducción se prueba que $\bigwedge n\, F(n) \in A^*$ y, más concretamente, que F es un camino en A.

$2 \Rightarrow 3$ es inmediato. Falta probar que $3 \Rightarrow 1$, pero, por el teorema 7.40, basta probar (suponiendo 3) que si $F : \mathbb{N} \longrightarrow \mathbb{N}$ es una aplicación inyectiva, existe el rango $F[\mathbb{N}]$. Para ello definimos el conjunto A de modo que $s \in A$ si y sólo si:

1. $\bigvee l(l = \ell(s) \wedge \bigwedge ij < l(F(i) = j \leftrightarrow s_j = i + 1))$,
2. $\bigvee l(l = \ell(s) \wedge \bigwedge j < l(s_j > 0 \rightarrow F(s_j \dot{-} 1) = n))$.

Claramente, las dos condiciones son de tipo Σ^0_1, pero también son de tipo Π^0_1, pues equivalen a

1. $\bigwedge l(l = \ell(s) \rightarrow \bigwedge ij < l(F(i) = j \leftrightarrow s_j = i + 1))$,
2. $\bigwedge l(l = \ell(s) \rightarrow \bigwedge j < l(s_j > 0 \rightarrow F(s_j \dot{-} 1) = i))$.

Por lo tanto, el esquema de Δ^0_1-comprensión garantiza la existencia de A. Una vez justificado esto, podemos escribir las condiciones como

1. $\bigwedge ij < \ell(s)(F(i) = j \leftrightarrow s_j = i + 1)$,
2. $\bigwedge j < \ell(s)(s_j > 0 \rightarrow F(s_j \dot{-} 1) = i)$.

Es inmediato que A es un árbol. Además, cada nodo $s \in A$ tiene a lo sumo dos sucesores, pues la segunda condición implica que si $\ell(s) = l$ y $t = s^\frown\langle n\rangle \in A$ (con lo que $n = t_l$), necesariamente $n = 0$ o bien $F(n \dot- 1) = l$, y la segunda condición la cumple a lo sumo un n, ya que si $F(n_1 \dot- 1) = l = F(n_2 \dot- 1)$, la inyectividad de F nos da que $n_1 \dot- 1 = n_2 \dot- 1$, luego $n_1 = n_2$.

Veamos ahora que A es infinito, para lo cual fijamos $k \in \mathbb{N}$. Por el principio de Σ^0_1-especificación acotada (teorema 7.26) existe el conjunto

$$Y = \{j < k \bigvee i\, F(i) = j\}.$$

A su vez, por Δ^0_1-comprensión, definimos el conjunto

$$f = \{n \mid \bigvee j < k((j \notin Y \wedge n = \langle j, 0\rangle_2) \vee (j \in Y \wedge \bigvee i < n\, n = \langle j, i+1\rangle_2))\}.$$

Es claro entonces que $f : I_k \longrightarrow \mathbb{N}$ es un conjunto finito, luego su código $[f]$ es una aplicación en sentido aritmético, para la cual está definido su rango (definición 2.19) y, en virtud de la correspondencia descrita en la página 96, se corresponde con un s tal que $\ell(s) = k$ y $\bigwedge j < k\, s_j = f(j)$. Explícitamente, tenemos que si $j < k$ no está en Y, entonces $s_j = 0$, mientras que si $j \in Y$, entonces $s_j \neq 0$ y $F(s_j \dot- 1) = j$. Esto implica que $s \in A$. Por lo tanto, A contiene nodos de cualquier altura k, lo que implica que A es infinito.

Por hipótesis, A tiene un camino $G : \mathbb{N} \longrightarrow A$. La condición 1. de la definición de A nos da que

$$\bigwedge ij(F(i) = j \leftrightarrow G(j) = i + 1).$$

Ahora basta tomar $R = \{j \mid G(j) > 0\}$, y así es claro que

$$\bigwedge j(\bigvee i\, F(i) = j \leftrightarrow j \in R),$$

es decir, que R es el rango de F. ∎

En particular en ACA_0 se puede probar el lema de König débil, lo que a su vez implica que todo teorema de LKD_0 es un teorema de ACA_0.

La no constructividad del teorema de König La prueba de $3 \Rightarrow 1$ en el teorema anterior es un ejercicio de ingenio, pero el lector haría bien en reflexionar sobre la prueba de $1 \Rightarrow 2$, pues es un ejemplo arquetípico de argumento no constructivo intuitivamente evidente. Aunque partamos de un árbol A tal que dispongamos de un algoritmo para determinar si un número arbirtrario s es o no un nodo de A y sepamos que A es infinito, pero finitamente ramificado, podemos asegurar que A tiene un camino, pero no tenemos garantías de que podamos determinar un camino concreto, en el sentido de que tengamos un algoritmo para determinar si un nodo arbitrario s está o no en el camino.

La prueba de la existencia de un camino se basa en que si sabemos que un nodo $s \in A$ tiene infinitos nodos por encima, alguno de sus sucesores inmediatos (que son un número finito) tiene que tener también infinitos nodos por encima, pero, en general, no tenemos un procedimiento para determinar cuáles de ellos nos sirven. Podemos asegurar que es verdad que alguno de sus sucesores tendrá infinitos nodos por encima, pero no podemos saber cuál.

En principio, sabemos que la consistencia de $\mathrm{I}\Sigma_1$ (o la de ARP, en su versión original) implica la de ACR_0 y a su vez la de LKD_0, pero este argumento implica que, no sólo LKD_0 es consistente, sino que todos sus teoremas son verdaderos en un sentido que todavía no estamos en condiciones de precisar formalmente, pero que consiste en que si en LKD_0 se demuestra un "$\bigvee X$", es cierto que existe un conjunto de números naturales que cumple lo que se indica, y si se demuestra un "$\bigwedge X$", lo que se afirma será cierto para cualquier conjunto de números naturales que podamos considerar. ∎

Para terminar con los hechos básicos sobre ACA_0, demostramos lo siguiente:

Teorema 7.42 ACA_0 *es finitamente axiomatizable.*

Demostración: Vamos a probar que las sentencias siguientes (todas las cuales son teoremas de ACA_0) permiten probar en ACA_0 sin el axioma de comprensión aritmética todos los casos particulares de dicho esquema, luego añadidos a los axiomas restantes de ACA_0 constituyen una axiomatización finita de esta teoría.[16] Notemos que del primero de ellos se deduce que

$$\langle x, y\rangle \equiv z|2z = (x+y)(x+y+1) + 2x$$

es una descripción propia. Escribiremos

$$\langle x\rangle \equiv x, \qquad \langle x_1, \ldots, x_n\rangle \equiv \langle \langle x_1, \ldots, x_{n-1}\rangle, x_n\rangle .$$

Así, por ejemplo, $\langle x, y, z\rangle$ es una abreviatura por $\langle \langle x, y\rangle, z\rangle$.

Par 1	$\bigwedge xy\overset{1}{\bigvee} z\ 2z = (x+y)(x+y+1) + 2x$
Par 2	$\bigwedge xyzw(\langle x, y\rangle = \langle z, w\rangle \to x = z \wedge y = w)$
Conjunto unitario	$\bigwedge x\bigvee A\bigwedge n(n \in A \leftrightarrow n = x)$
Sucesor	$\bigvee A\bigwedge xy(\langle x, y\rangle \in A \leftrightarrow y = x')$
Suma	$\bigvee A\bigwedge xyz(\langle x, y, z\rangle \in A \leftrightarrow x = y + z)$
Producto	$\bigvee A\bigwedge xyz(\langle x, y, z\rangle \in A \leftrightarrow x = yz)$
Intersección	$\bigwedge XY\bigvee Z\bigwedge u(u \in Z \leftrightarrow u \in X \wedge u \in Y)$
Complemento	$\bigwedge X\bigvee Y\bigwedge u(u \in Y \leftrightarrow u \notin X)$
Dominio	$\bigwedge A\bigvee B\bigwedge x(x \in B \leftrightarrow \bigvee y\ \langle x, y\rangle \in A)$
Producto cartesiano	$\bigwedge A\bigvee B\bigwedge xy(\langle x, y\rangle \in B \leftrightarrow x \in A)$
Relación inversa	$\bigwedge A\bigvee B\bigwedge xy(\langle x, y\rangle \in B \leftrightarrow \langle y, x\rangle \in A)$
Identidad	$\bigvee A\bigwedge xy(\langle x, y\rangle \in A \leftrightarrow x = y)$
Permutación	$\bigwedge A\bigvee B\bigwedge xyz(\langle x, y, z\rangle \in B \leftrightarrow \langle y, z, x\rangle \in A)$
Permutación	$\bigwedge A\bigvee B\bigwedge xyz(\langle x, y, z\rangle \in B \leftrightarrow \langle x, z, y\rangle \in A)$
Cuádrupla	$\bigwedge A\bigvee B\bigwedge xyzw(\langle x, y, z, w\rangle \in B \leftrightarrow \langle y, z, w\rangle \in A)$

Es inmediato que todas estas sentencias son teoremas de ACA_0. Las dos primeras porque son traducciones de teoremas de AP, y las demás se siguen

[16] Alternativamente, podemos tomar como axiomas todos los axiomas de ACA_0 menos el esquema de comprensión aritmética, más el conjunto finito casos particulares de dicho esquema necesarios para demostrar las sentencias anteriores.

fácilmente del esquema de comprensión aritmética. Por ejemplo, para probar el axioma de la relación inversa basta tomar

$$B \equiv \{n \mid \bigvee xy(n = \langle x, y\rangle \wedge \langle y, x\rangle \in A)\}.$$

Ahora vamos a ver que con estos axiomas es posible demostrar el esquema de comprensión aritmética. En primer lugar observamos que los axiomas de la intersección y el complemento nos permiten definir $A \cap B$, $A \cup B$, $\mathbb{N}$, $\mathbb{N} \setminus A$ y $\varnothing$.

De los axiomas del par se puede deducir cualquier caso particular de los esquemas siguientes:

$$\bigwedge x_1 \cdots x_n y_1 \cdots y_n(\langle x_1, \ldots, x_n\rangle = \langle y_1, \ldots, y_n\rangle \to x_1 = y_1 \wedge \cdots \wedge x_n = y_n)$$

$$\bigwedge x_1 \cdots x_{n+p}(\langle \langle x_1, \ldots, x_n\rangle, x_{n+1}, \ldots, x_{n+p}\rangle = \langle x_1, \ldots, x_{n+p}\rangle).$$

El axioma de extensionalidad nos da que el conjunto cuya existencia postula el axioma del dominio es único, por lo que podemos definir

$$\mathcal{D}A \equiv B|\bigwedge x(x \in B \leftrightarrow \bigvee y\ \langle x, y\rangle \in A).$$

Aplicando a $\mathcal{D}A$ el axioma de la relación inversa (junto con el axioma de extensionalidad) obtenemos que

$$\bigwedge A \overset{1}{\bigvee} B \bigwedge x(x \in B \leftrightarrow \bigvee y\ \langle y, x\rangle \in A),$$

lo que nos permite definir

$$\mathcal{R}A \equiv B|\bigwedge x(x \in B \leftrightarrow \bigvee y\ \langle y, x\rangle \in A).$$

Enumeramos a continuación algunos resultados más:

1) $\bigwedge A \bigvee B \bigwedge xy(\langle x, y\rangle \in B \leftrightarrow y \in A)$.

   Por el axioma de la relación inversa al axioma del producto cartesiano.

2a) $\bigwedge A \bigvee B \bigwedge xyz(\langle x, y, z\rangle \in B \leftrightarrow \langle x, y\rangle \in A)$

2b) $\bigwedge A \bigvee B \bigwedge xyz(\langle x, z, y\rangle \in B \leftrightarrow \langle x, y\rangle \in A)$

2c) $\bigwedge A \bigvee B \bigwedge xyz(\langle z, x, y\rangle \in B \leftrightarrow \langle x, y\rangle \in A)$

   Por el axioma del producto cartesiano $\bigwedge A \bigvee B \bigwedge wz(\langle w, z\rangle \in B \leftrightarrow w \in A)$. Haciendo $w = \langle x, y\rangle$ tenemos a), y aplicando los axiomas de permutación obtenemos b) y c).

3) $\bigwedge A \bigvee B \bigwedge x_1 \cdots x_n y(\langle x_1, \ldots, x_n, y\rangle \in B \leftrightarrow \langle x_1, \ldots, x_n\rangle \in A)$.

   Por el axioma del producto cartesiano, haciendo $x = \langle x_1, \ldots, x_n\rangle$.

4) $\bigwedge A \bigvee B \bigwedge x_1 \cdots x_n y_1 \cdots y_k(\langle x_1, \ldots, x_n, y_1, \ldots, y_k\rangle \in B \leftrightarrow \langle x_1, \ldots, x_n\rangle \in A)$

   Por 3) aplicado k veces.

5) $\bigwedge A \bigvee B \bigwedge x_1 \cdots x_n y(\langle x_1, \ldots, x_{n-1}, y, x_n\rangle \in B \leftrightarrow \langle x_1, \ldots, x_n\rangle \in A)$
Por 2b)

6) $\bigwedge A \bigvee B \bigwedge y_1 \cdots y_k x_1 x_2(\langle y_1, \ldots, y_k, x_1, x_2\rangle \in B \leftrightarrow \langle x_1, x_2\rangle \in A)$
Por 2c).

Veamos ahora que, si $t(x_1, \ldots, x_n)$ es un término aritmético de primer orden sin descriptores cuyas variables libres están entre las indicadas, podemos probar (por inducción sobre la longitud de t) que

$$\bigvee A \bigwedge x_1 \cdots x_n y(\langle x_1, \ldots, x_n, y\rangle \in A \leftrightarrow y = t(x_1, \ldots, x_n)).$$

En efecto, si $t \equiv x_i$ por el axioma de la identidad existe un B_1 tal que

$$\bigwedge x_i y(\langle x_i, y\rangle \in B_1 \leftrightarrow x_i = y).$$

Si $i > 1$ aplicamos 6) para concluir que existe un B_2 tal que

$$\bigwedge x_1 \cdots x_i y(\langle x_1, \ldots, x_i, y\rangle \in B_2 \leftrightarrow x_i = y).$$

Si $i < n$ aplicamos 5) varias veces para concluir que existe un A tal que

$$\bigwedge x_1 \cdots x_n y(\langle x_1, \ldots, x_n, y\rangle \in A \leftrightarrow x_i = y).$$

Si $t \equiv 0$, por el axioma del conjunto unitario $\bigvee B \bigwedge y(y \in B \leftrightarrow y = 0)$. Basta aplicar 1) con $x = \langle x_1, \ldots, x_n\rangle$.

Si $t \equiv t_0'$, por hipótesis de inducción existe un conjunto B tal que

$$\bigwedge x_1 \cdots x_n z(\langle x_1, \ldots, x_n, z\rangle \in B \leftrightarrow z = t_0(x_1, \ldots, x_n)).$$

Por otro lado, por el axioma del sucesor y de la relación inversa, existe un C tal que

$$\bigwedge yz(\langle y, z\rangle \in C \leftrightarrow y = z').$$

Por 5) existe B' tal que

$$\bigwedge x_1 \cdots x_n yz(\langle x_1, \ldots, x_n, y, z\rangle \in B' \leftrightarrow \langle x_1, \ldots, x_n, z\rangle \in B).$$

Por 6) existe C' tal que

$$\bigwedge x_1 \cdots x_n yz(\langle x_1, \ldots, x_n, y, z\rangle \in C' \leftrightarrow \langle y, z\rangle \in C).$$

Basta tomar $A \equiv \mathfrak{D}(B' \cap C')$.

Supongamos ahora que $t \equiv t_1 + t_2$. Sean B_1 y B_2 tales que

$$\bigwedge x_1 \cdots x_n z(\langle x_1, \ldots, x_n, z\rangle \in B_1 \leftrightarrow z = t_1(x_1, \ldots, x_n)),$$

$$\bigwedge x_1 \cdots x_n z(\langle x_1, \ldots, x_n, z\rangle \in B_2 \leftrightarrow z = t_2(x_1, \ldots, x_n)).$$

Por el axioma de la suma existe un C tal que

$$\bigwedge xz_1z_2(\langle y, z_1, z_2\rangle \in C \leftrightarrow y = z_1 + z_2).$$

Sean B_1' y B_2' tales que

$$\bigwedge x_1 \cdots x_n y z_1 z_2(\langle x_1, \ldots, x_n, y, z_1, z_2\rangle \in B_1' \leftrightarrow \langle x_1, \ldots, x_n, z_1\rangle \in B_1)$$

$$\bigwedge x_1 \cdots x_n y z_1 z_2(\langle x_1, \ldots, x_n, y, z_1, z_2\rangle \in B_2' \leftrightarrow \langle x_1, \ldots, x_n, z_2\rangle \in B_2)$$

La existencia de B_1' se sigue del axioma del producto cartesiano aplicado al conjunto dado por 5). La existencia de B_2' se sigue de aplicar 5) dos veces. Por el axioma de la cuádrupla existe C' tal que

$$\bigwedge x_1 \cdots x_n y z_1 z_2(\langle x_1, \ldots, x_n, y, z_1, z_2\rangle \in C' \leftrightarrow \langle y, z_1, z_2\rangle \in C).$$

Ahora basta tomar $A = \mathcal{D}\mathcal{D}(B_1' \cap B_2' \cap C')$.

El caso $t \equiv t_1 \cdot t_2$ es análogo. Veamos ahora que si $\phi(x_1, \ldots, x_n, X_1, \ldots, X_r)$ es una fórmula aritmética sin descriptores se puede probar

$$\bigwedge X_1 \cdots X_r \bigvee A \bigwedge x_1 \cdots x_n(\langle x_1 \ldots, x_n\rangle \in A \leftrightarrow \phi(x_1, \ldots, x_n, X_1, \ldots, X_n)).$$

En efecto, si $\phi \equiv t_1 = t_2$ hemos probado que existen conjuntos B_1 y B_2 tales que

$$\bigwedge x_1 \cdots x_n y(\langle x_1, \ldots, x_n, y\rangle \in B_1 \leftrightarrow y = t_1),$$
$$\bigwedge x_1 \cdots x_n y(\langle x_1, \ldots, x_n, y\rangle \in B_2 \leftrightarrow y = t_2).$$

Basta tomar $A = \mathcal{D}(B_1 \cap B_2)$.

Si $\phi \equiv t \in X$, sabemos que existe un B tal que

$$\bigwedge x_1 \cdots x_n y(\langle x_1, \ldots, x_n, y\rangle \in B \leftrightarrow y = t),$$

y por 1) existe un C tal que

$$\bigwedge x_1 \cdots x_n y(\langle x_1, \ldots, x_n, y\rangle \in C \leftrightarrow y \in X).$$

Basta tomar $A = \mathcal{D}(B \cap C)$.

Si $\phi \equiv X = Y$, basta tomar $A = \mathbb{N}$ o bien $A = \varnothing$.

Si el resultado vale para α y β y A_1 y A_2 son los conjuntos correspondientes, entonces $\mathbb{N} \setminus A_1$ cumple el resultado para $\neg\alpha$, mientras que $(\mathbb{N} \setminus A_1) \cup A_2$ lo cumple para $\alpha \to \beta$.

Supongamos finalmente que $\phi \equiv \bigwedge x\psi$. Por hipótesis de inducción existe un conjunto B tal que

$$\bigwedge x_1 \cdots x_n x(\langle x_1, \ldots, x_n, x\rangle \in B \leftrightarrow \psi(x_1, \ldots, x_n, x)).$$

Basta tomar $A = \mathbb{N} \setminus \mathcal{D}(\mathbb{N} \setminus B)$.

Finalmente observamos que, para toda fórmula aritmética sin descriptores $\phi(x, x_1, \ldots, x_s, X_1, \ldots, X_r)$, se cumple

$$\bigwedge X_1 \cdots X_r \bigwedge x_1 \cdots x_s \bigvee X \bigwedge x(x \in X \leftrightarrow \phi(x)).$$

En efecto, hemos probado que existe un B tal que

$$\bigwedge x_1 \cdots x_s x(\langle x_1, \ldots, x_s, x\rangle \in B \leftrightarrow \phi(x, x_1, \ldots, x_n)).$$

Por el axioma del conjunto unitario existe un C tal que

$$\bigwedge x(x \in C \leftrightarrow x = \langle x_1, \ldots, x_s\rangle).$$

Por el axioma de producto cartesiano existe un C' tal que

$$\bigwedge yx(\langle y, x\rangle \in C' \leftrightarrow y = \langle x_1, \ldots, x_n\rangle).$$

Basta tomar $X = \mathcal{R}(B \cap C')$, pues entonces

$$x \in X \leftrightarrow \bigvee y\ \langle y, x\rangle \in B \cap C' \leftrightarrow \langle x_1, \ldots, x_n, x\rangle \in B \leftrightarrow \phi(x, x_1, \ldots, x_n).$$

■

7.7 Recursión aritmética

En ACR_0 hemos probado el teorema 7.17, que nos garantiza que podemos definir funciones $F : X \times \mathbb{N} \longrightarrow \mathbb{N}$ por recursión completa, es decir, que podemos definir $F(x, n)$ supuesto definido $F(x, m)$ para todo $m < n$. Informalmente, también podemos definir recurrentemente sucesiones de conjuntos $\{Y_i\}_{i=0}^{\infty}$, es decir, que podemos definir Y_i supuesto definido $\{Y_j\}_{j<i}$, pero esta forma de definir conjuntos no es formalizable ni siquiera en ACA_0. Vamos a analizar la situación.

Fijemos una fórmula aritmética $\phi(n, Z)$, que puede tener más parámetros de primer y segundo orden, además de las dos variables indicadas. Fijados dichos parámetros, la fórmula ϕ determina una forma de construir un conjunto $\Phi(Z)$ a partir de cualquier conjunto dado Z, a saber, el conjunto

$$\Phi(Z) = \{n \mid \phi(n, Z)\}.$$

Tenemos así una "aplicación" metamatemática que asigna conjuntos a conjuntos, y vamos a tratar de usarla para definir una sucesión de conjuntos $\{Y_i\}_{i=0}^{\infty}$.

Ante todo, técnicamente no podemos hablar de una sucesión de conjuntos $\{Y_i\}_{i=0}^{\infty}$ entendida como una aplicación que a cada i le asigna un conjunto Y_i, pero podemos formalizarla como un conjunto Y tal que $Y_i = \{n \mid \langle i, n\rangle_2 \in Y\}$. La idea es definir $Y_i = \Phi(Y^i)$, donde

$$Y^i = (\{0\} \times Y_0) \cup \cdots \cup (\{i-1\} \times Y_{i-1}) = \{Y_j\}_{j<i},$$

de modo que la fórmula ϕ es la que determina cómo se construye Y_i en función de los términos precedentes de la sucesión. En particular tenemos que $Y^0 = \varnothing$, que es un conjunto aritmético definido por la fórmula $\bar{\phi}_0(z) \equiv z \neq z$, y el primer término de la sucesión que queremos definir será

$$Y_0 = \{n \mid \phi(n, \varnothing)\},$$

que es un conjunto aritmético. Más explícitamente, si llamamos $\phi_0(n)$ a la fórmula[17] que resulta de sustituir cada subfórmula $i \in Z$ de $\phi(n, Z)$ por $i \neq i$ (la fórmula que define a $Y^0 = \varnothing$) y hacemos

$$\phi^1(z) \equiv \bigvee n(z = \langle 0, n\rangle_2 \wedge \phi_0(n)),$$

tenemos que $\phi^1(z)$ es una fórmula aritmética y en ACA_0 podemos definir

$$Y_0 = \{n \mid \phi_0(n)\}, \qquad Y^1 = Y^0 \cup (\{0\} \times Y_0) = \{z \mid \phi^1(z)\}.$$

A su vez, podemos definir

$$Y_1 = \Phi(Y^1) = \{n \mid \phi(n, Y^1)\}, \qquad Y^2 = Y^1 \cup (\{1\} \times Y_1).$$

Si llamamos $\phi_1(n)$ a la fórmula aritmética que resulta de sustituir en $\phi(n, Z)$ cada subfórmula $z \in Z$ por $\phi^1(z)$ (la fórmula que define a Y^1) y

$$\phi^2(z) \equiv \phi^1(z) \vee \bigvee n(z = \langle 1, n\rangle_2 \wedge \phi_1(n)),$$

tenemos que

$$Y_1 = \{z \mid \phi_1(z)\}, \qquad Y^2 = \{z \mid \phi^2(z)\}.$$

A su vez, podemos definir

$$Y_2 = \Phi(Y^2) = \{n \mid \phi(n, Y^2)\}, \qquad Y^3 = Y^2 \cup (\{2\} \times Y_2).$$

En general, para cada numeral $0^{(n)}$, en ACA_0 podemos definir recurrentemente las sucesiones

$$Y^0 = \varnothing, \qquad Y_{0^{(n+1)}} = \Phi(Y^{0^{(n)}}), \qquad Y^{0^{(n+1)}} = Y^{0^{(n)}} \cup (\{0^{(n)}\} \times Y_{0^{(n)}}).$$

Es importante destacar que podemos definir los conjuntos $Y_0, Y_1, Y_2, \ldots$, así como los conjuntos $Y^0, Y^1, Y^2, \ldots$, pero no tenemos expresiones generales Y_n o Y^n, donde n es una variable.[18]

[17] Podemos suponer que la fórmula $\phi(n, Z)$ no contiene subfórmulas de tipo $U = V$, porque siempre pueden sustituirse por $\bigwedge u(u \in U \leftrightarrow u \in V)$ y el resultado es una fórmula lógicamente equivalente.

[18] Una situación análoga es que, en ZF, podemos definir $0 = \varnothing$, $1 = \{0\}$, $2 = \{0, 1\}$, $3 = \{0, 1, 2\}, \ldots$ y así tenemos infinitas definiciones, pero éstas no equivalen a la definición de una fórmula "n es un número natural". Para definir esta fórmula hacen falta ideas nuevas que no están presentes en las infinitas definiciones anteriores. Nosotros tenemos también infinitas definiciones, y nos gustaría tener una idea análoga a la que en ZF permite definir la fórmula "n es un número natural".

El razonamiento precedente muestra que los conjuntos $Y_{0^{(n)}}$, $Y^{0^{(n)}}$ son aritméticos, pero cada uno de ellos está definido por una fórmula aritmética diferente, $\phi_0(n), \phi_1(n), \phi_2(n), \ldots$ y $\phi^0(z), \phi^1(z), \phi^2(z), \ldots$, cada una más complicada que la anterior, por lo que, en general, no tenemos una forma de definir en ACA_0 la sucesión

$$Y = \{Y_i\}_{i=0}^{\infty} = \{\langle i, n\rangle_2 \mid n \in Y_i\}.$$

No obstante, podemos describir Y mediante una fórmula. Para ello definimos $I_i = \{j \mid j < i\}$ y, para cualquier conjunto Z:

$$Z_i = \{n \mid \langle i, n\rangle_2 \in Z\},$$

$$Z^i = Z \cap (I_i \times \mathbb{N}) = \{z \mid z \in Z \wedge \bigvee jn(j < i \wedge z = \langle i, n\rangle_2)\},$$

Si llamamos $\phi^*(n, Z, i)$ a la fórmula (aritmética) que resulta de sustituir cada subfórmula $z \in Z$ de $\phi(n, Z)$ por $z \in Z \wedge \bigvee jn(j < i \wedge z = \langle i, n\rangle_2)$, en ACR_0 podemos probar:

$$\phi^*(n, Z, i) \leftrightarrow \phi(n, Z^i).$$

Así, el conjunto Y que queremos definir está caracterizado por la fórmula

$$Y \subset \mathbb{N} \times \mathbb{N} \wedge \bigwedge i\ Y_i = \Phi(Y^i),$$

que equivale a la fórmula aritmética

$$\mathrm{RA}_\phi(Y) \equiv \bigwedge z(z \in Y \leftrightarrow \bigvee in(z = \langle i, n\rangle_2 \wedge \phi^*(n, Y, i))).$$

El problema es que en ACA_0 no podemos demostrar que exista un conjunto Y que cumpla esta fórmula. Más detalladamente, si definimos

$$\mathrm{RA}^k_\phi(Y) \equiv \bigwedge z(z \in Y \leftrightarrow \bigvee in(z = \langle i, n\rangle_2 \wedge i < k \wedge \phi^*(n, Y, i))),$$

podemos demostrar que existen conjuntos aritméticos $Y^0, Y^1, Y^2, \ldots$, que cumplen las fórmulas

$$\mathrm{RA}^0_\phi(Y^0), \quad \mathrm{RA}^1_\phi(Y^1), \quad \mathrm{RA}^2_\phi(Y^2), \quad \ldots$$

pero no podemos demostrar que $\bigwedge k \bigvee Y\, \mathrm{RA}^k_\phi(Y)$. Similarmente, las fórmulas

$$\mathrm{RA}_\phi(Z, k) \equiv \bigwedge n(n \in Z \leftrightarrow \bigvee Y(\mathrm{RA}^{k+1}_\phi(Y) \wedge \langle k, n\rangle_2 \in Y))$$

caracterizan los conjuntos Y_k, es decir, que podemos demostrar que existen conjuntos aritméticos $Y_0, Y_1, Y_2, \ldots$, que cumplen las fórmulas

$$\mathrm{RA}_\phi(Y_0, 0), \quad \mathrm{RA}_\phi(Y_1, 1), \quad \mathrm{RA}_\phi(Y_2, 2), \quad \ldots$$

pero no podemos demostrar que $\bigwedge k \bigvee Y\, \mathrm{RA}_\phi(Y, k)$.

La restricción del axioma de comprensión a fórmulas aritméticas no es caprichosa, sino que nos garantiza que los axiomas de ACA_0 tienen una interpretación

precisa (y verdadera): sabemos lo que decimos cuando afirmamos que una propiedad se cumple para todos los números naturales, o que existe uno que la cumple, aunque no tengamos un procedimiento explícito para comprobar si es así. Tal vez no sepamos si un número natural pertenece o no a un conjunto definido mediante una fórmula aritmética, pero sabemos lo que significa que así sea. No podríamos decir lo mismo en general si admitimos fórmulas no aritméticas en la definición de un conjunto: no podemos precisar qué queremos decir cuando afirmamos que todos los conjuntos de números naturales cumplen una propiedad, o que existe uno que la cumple, a menos que tengamos un razonamiento que lo justifique, que en el segundo caso puede reducirse a encontrar un ejemplo explícito.

Pero en el caso concreto de $\mathrm{RA}_\phi(Y)$, fijado un número natural z, sí que sabemos lo que significa $\bigvee Y(\mathrm{RA}_\phi(Y) \wedge 0^{(z)} \in Y)$. El número z será de la forma $z = \langle i, n\rangle_2$ y lo que estamos diciendo es que n está en un conjunto $Y_{0^{(i)}}$ que podemos definir mediante una fórmula aritmética explícita. Puede que no sepamos comprobar si es así o no, pero sabemos lo que esto significa. De hecho, en ACR_0 podemos probar que, si existe Y, es único:

Teorema 7.43 *Para cada fórmula aritmética $\phi(n, Z)$, la fórmula siguiente se demuestra en* ACR_0*:*

$$\bigwedge YY'(\mathrm{RA}_\phi(Y) \wedge \mathrm{RA}_\phi(Y) \to Y = Y').$$

Demostración: Supongamos que se cumple $\mathrm{RA}_\phi(Y)$ y $\mathrm{RA}_\phi(Y')$, pero que $Y \neq Y'$. Entonces existe un $z \in (Y \setminus Y') \cup (Y' \setminus Y)$. Pongamos que $z = \langle i, n\rangle_2$. Entonces podemos tomar el mínimo i tal que

$$\bigvee zn(z = \langle i, n\rangle_2 \wedge z \in (Y \setminus Y') \cup (Y' \setminus Y)).$$

Notemos que esta fórmula es Σ^0_1, luego el teorema 7.12 nos asegura la existencia del mínimo. Tenemos entonces que existe un n tal que $\langle i, n\rangle_2 \in (Y \setminus Y') \cup (Y' \setminus Y)$, pero

$$\bigwedge j < i \bigwedge m(\langle j, m\rangle_2 \in Y \leftrightarrow \langle j, m\rangle_2 \in Y').$$

No perdemos generalidad si suponemos que $\langle i, n\rangle_2 \in Y \setminus Y'$. Por la definición de $\mathrm{RA}_\phi(Y)$, tenemos que se cumple $\phi^*(n, Y, i)$, lo cual equivale a $\phi(n, Y^i)$, o también a $\phi(n, Y'^i)$, o también a $\phi^*(n, Y', i)$, pero entonces tendría que ser $(i, n) \in Y'$, y tenemos una contradicción. ∎

Definición 7.44 Llamaremos RA_0 a la teoría axiomática que resulta de añadir a ACR_0 los axiomas

$$\mathrm{RA}(\phi) \equiv \bigvee Y\, \mathrm{RA}_\phi(Y)$$

para toda fórmula aritmética $\phi(n, Z)$, tal vez con más variables libres.

Observemos que en RA_0 podemos probar el axioma de comprensión aritmética, con lo que, en realidad, RA_0 extiende a ACA_0. En efecto, si $\phi(x)$ es una fórmula aritmética el axioma $\mathrm{RA}(\phi)$ (tomando como Y cualquier variable que

no esté en ϕ) nos da un conjunto Y con la propiedad de que $\langle 0, n\rangle_2 \in Y \leftrightarrow \phi(n)$, luego el conjunto

$$\{n \mid \langle 0, n\rangle \in Y\},$$

definible en ACR_0, es el conjunto cuya existencia postula el axioma de comprensión aritmética para la fórmula dada.

En suma, RA_0 resulta de añadir a ACA_0 la posibilidad de definir sucesiones de conjuntos $Y = \{Y_i\}_{i=0}^{\infty}$ por recursión completa mediante fórmulas aritméticas, de modo que cada Y_i está definido a partir de la sucesión $Y^i = \{Y_j\}_{j<i}$ mediante una fórmula aritmética:

$$Y_i = \{n \mid \phi(n, Y^i)\} = \Phi(\{Y_j\}_{j<i}).$$

Hemos visto que, aunque técnicamente la definición de la sucesión Y involucra particularizadores de segundo orden, en realidad no estamos planteando la existencia de un conjunto sin referencia alguna sobre cuál podría ser, sino que los conjuntos que consideramos que "existen" no son sino las sucesiones de términos previos de la sucesión. Más concretamente: Y_0 tiene una definición aritmética explícita, Y_1 se define aritméticamente admitiendo la existencia de Y_0, que ya está justificada, Y_2 se define aritméticamente admitiendo la existencia de Y_0, Y_1, que ya está justificada, etc., por lo que podemos afirmar que todos los conjuntos $Y_0, Y_1, Y_2, \ldots$ están bien definidos informalmente, luego también lo está la sucesión de todos ellos. El axioma de Recursión Aritmética no hace más que formalizar este argumento válido informalmente, por lo que todos los teoremas de RA_0 determinan argumentos informales válidos.

En RA_0 desaparece la distinción entre premodelos y modelos que tuvimos que introducir en 7.29 para trabajar en ACR_0:

Teorema 7.45 ($\mathbf{RA_0}$) *Todo premodelo de un lenguaje formal se extiende de forma única a un modelo.*

Demostración: Dado un premodelo M de un lenguaje formal $\mathcal{L}$, tenemos que definir la función $M_3 : P_M \longrightarrow M \cup \{0, 1\}$ que requiere la definición de modelo o, equivalentemente, las expresiones $M(t)[v]$ y $M \vDash \alpha[v]$. Para ello definiremos por recursión aritmética una sucesión $\{F_\theta\}_{\theta=0}^{\infty}$ de modo que:

1. Si θ es un semitérmino de $\mathcal{L}$, entonces $F_\theta : \mathbb{N} \longrightarrow M$ es la función que a cada valoración v de θ le hace corresponder $F_\theta(v) = M(\theta)[v]$ y, si v no es una valoación de θ, entonces $F_\theta(v) = 0$.

2. Si θ es una semifórmula de $\mathcal{L}$, entonces $F_\theta : \mathbb{N} \longrightarrow \{0, 1\}$ es la función tal que si v es una valoración de θ tal que $M \vDash \theta[v]$, entonces $F_\theta(v) = 1$, y en caso contrario $F_\theta(v) = 0$.

3. SI θ no es una semiexpresión de $\mathcal{L}$, entonces $F_\theta = \varnothing$.

Basta observar que las condiciones que debe cumplir la sucesión $\{F_\theta\}_{\theta=0}^{\infty}$ para que la función $M_3 : P_M \longrightarrow M \cup \{0, 1\}$ dada por $M_3(\theta, v) = F_\theta(v)$ cumpla la definición de modelo permiten definir aritméticamente cada F_θ a partir de M y de la sucesión $\{F_{\theta'}\}_{\theta'<\theta}$. ∎

El modelo natural de AP y de ARP En particular, este teorema se aplica al premodelo natural $\mathbb{N}$ de la aritmética de Peano, que en AR_0 determina un modelo, de modo que tenemos definidas las expresiones $\mathbb{N}(t)[v]$ y $\mathbb{N} \vDash \alpha[v]$, de modo que todos los axiomas de Peano (luego todos los teoremas de AP) cumplen $\mathbb{N} \vDash \alpha$.

El teorema de completitud semántica implica entonces que

$$\underset{\mathrm{AR}_0}{\vdash} \mathrm{Consis\,AP}.$$

En otras palabras: en AR_0 podemos formalizar la demostración obvia de que la aritmética de Peano es consistente, a saber, la que se basa en que tiene por modelo al conjunto de los números naturales con las operaciones aritméticas usuales.

Más aún, también podemos considerar a $\mathbb{N}$ como modelo de ARP interpretando cada funtor f de $\mathcal{L}_{\mathrm{arp}}$ como la función recursiva primitiva $F(f)$ definida en 1.2. Es fácil ver entonces que el número $N(t)[v]$ denotado por un término de $\mathcal{L}_{\mathrm{arp}}$ definido en 1.6 es el mismo $\mathbb{N}(t)[v]$ determinado por la definición de modelo.[19] Similarmente, una fórmula atómica α de $\mathcal{L}_{\mathrm{arp}}$ cumple $\vDash \alpha[v]$ en el sentido definido en 1.7 si y sólo si cumple $\mathbb{N} \vDash \alpha[v]$ en el sentido general de la teoría de modelos.

La demostración informal del teorema 1.16 puede verse ahora como una demostración formal en RA_0. Más aún, ahora podemos probar un teorema de completitud:

Teorema 7.46 ($\boldsymbol{\Sigma_1}$-completitud de ARP) *Si α es una sentencia Σ_1 de $\mathcal{L}_{\mathrm{arp}}$, entonces*

$$\mathbb{N} \vDash \alpha \quad \textit{si y sólo si} \quad \underset{\mathrm{ARP}}{\vdash} \alpha.$$

Demostración: Que $\underset{\mathrm{ARP}}{\vdash} \alpha$ implica que α es verdadera es un caso particular del teorema de corrección 1.12. La implicación relevante es la opuesta. Veámosla primero para sentencias Δ_0. Puesto que toda sentencia Δ_0 es equivalente en ARP a una sentencia atómica, podemos suponer que $\alpha \equiv t_1 = t_2$, para ciertos designadores t_1 y t_2.

Si $\mathbb{N} \vDash \alpha$, esto significa que $N(t_1) = N(t_2)$. Si llamamos n a este número natural, el teorema 1.16 nos da que $\underset{\mathrm{ARP}}{\vdash} t_1 = 0^{(n)}$ y $\underset{\mathrm{ARP}}{\vdash} t_2 = 0^{(n)}$, luego $\underset{\mathrm{ARP}}{\vdash} t_1 = t_2$, que es lo mismo que $\underset{\mathrm{ARP}}{\vdash} \alpha$.

Si $\alpha \equiv \bigvee u\, \beta(u)$ es de tipo Σ_1, entonces $\mathbb{N} \vDash \alpha$ significa que existe un número natural n tal que $\mathbb{N} \vDash \beta(0^{(n)})$, luego, por el caso ya probado, $\underset{\mathrm{ARP}}{\vdash} \beta(0^{(n)})$, de donde a su vez $\underset{\mathrm{ARP}}{\vdash} \alpha$. ∎

[19] Notemos que en 1.6 estamos considerando a $\mathcal{L}_{\mathrm{arp}}$ como el lenguaje específico sin cuantificadores de la aritmética recursiva primitiva, mientras que la definición de modelo se aplica a su extensión de primer orden, pero lo que decimos tiene sentido porque todo término del primer lenguaje lo es también de su extensión de primer orden.

Capítulo VIII

Teoría de la recursión

Ya hemos señalado en varias ocasiones la razón por la que limitar los principios de inducción y comprensión en ACR_0 respecto de ACA_0 no es un capricho, sino que nos permite distinguir los resultados que admiten pruebas constructivas de los que incluyen argumentos no constructivos. En este capítulo vamos a analizar la situación con más detalle. Para ello introduciremos el concepto general de "función recursiva". Las funciones recursivas son las funciones definidas sobre los números naturales que pueden calcularse mediante un algoritmo, es decir, mediante un programa de ordenador, con garantías de obtener en un tiempo finito el valor que toman para unos argumentos dados. A su vez, a partir de ellas podemos definir los conjuntos recursivos como los conjuntos de números naturales (o, más en general, de n-tuplas de números naturales) cuya función característica es recursiva, lo que equivale a que dispongamos de un algoritmo para determinar si un número (o n-tupla) está o no en el conjunto.

Hay muchos contextos en los que resulta natural exigir la recursividad de un conjunto. Por ejemplo, es razonable exigir que el conjunto de los axiomas de una teoría axiomática sea recursivo, pues, ¿de qué valdría una teoría axiomática en la que no podemos saber si una fórmula dada es o no un axioma? Y en el capítulo siguiente veremos algunos resultados fundamentales en los que hipótesis de este tipo son esenciales.

8.1 Funciones recursivas

Trabajamos en ACR_0. Definimos $\mathbb{N}^r \equiv \{s \mid \ell(s) = r\}$. Vamos a considerar ahora funciones de la forma $F : \mathbb{N}^r \longrightarrow \mathbb{N}$. En la práctica podemos pensar en ellas como funciones de r variables, pero en teoría sus argumentos son números naturales de longitud r. En la práctica escribiremos, por ejemplo, $F(x, y)$ en lugar de $F(\langle x, y\rangle)$.

Vamos a pasar por alto los tecnicismos derivados del hecho de que no es lo mismo $\mathbb{N}$ que $\mathbb{N}^1$, o de que no es lo mismo $\mathbb{N}^k \times \mathbb{N}^r$ que $\mathbb{N}^{k+r}$, aunque son esencialmente lo mismo. La relación es que podemos definir $\mathbb{N} \longrightarrow \mathbb{N}^1$ biyectiva mediante $n \mapsto \langle n\rangle$, y también $\mathbb{N}^k \times \mathbb{N}^r \longrightarrow \mathbb{N}^{k+r}$ mediante $\langle s, t\rangle \mapsto s^\frown t$.

Usando estas biyecciones podemos transformar un resultado que involucre unos conjuntos en otros.

Otro hecho técnico es que hemos definido las aplicaciones como conjuntos de pares $\langle x, y\rangle_2$, que no son los mismos que los pares $\langle x, y\rangle$.

Por ejemplo, a cada aplicación $F : \mathbb{N}^r \longrightarrow \mathbb{N}$ podemos asociarle su *gráfica*

$$G(F) = \{z \mid \bigvee sn \leq z\, z = s^\frown\langle n\rangle \wedge \langle s, n\rangle_2 \in F\} \subset \mathbb{N}^{r+1},$$

que no es exactamente el mismo conjunto que F, pero tenemos una biyección natural $F \longrightarrow G(F)$ dada por $\langle s, n\rangle_2 \mapsto s^\frown\langle n\rangle$.

En la práctica no nos preocuparemos de explicitar las "traducciones" necesarias que transforman unos de estos conjuntos en otros cuando sea necesario hacerlo.

Por ejemplo, recordemos que, si F es un conjunto, llamamos

$$F_i \equiv \{s \mid \langle i, s\rangle_2 \in F\}.$$

Si se cumple que $\bigwedge i < m\, F_i : \mathbb{N}^n \longrightarrow \mathbb{N}$, podemos definir una función

$$\langle F\rangle_m : \mathbb{N}^n \longrightarrow \mathbb{N}^m$$

tal que

$$\langle F\rangle_m (x) = \langle F_0(x), \ldots, F_{m-1}(x)\rangle\,.$$

Más precisamente, el teorema[1] 7.16 nos permite definir $G : \mathbb{N}^{n+1} \longrightarrow \mathbb{N}$ mediante

$$G(x, 0) = 0, \qquad G(x, i+1) = G(x, i)^\frown\langle F_i(x)\rangle\,,$$

de modo que una simple inducción sobre m prueba que

$$\ell(G(x, m)) = m \wedge \bigwedge i < m\, G(x, m)_i = F_i(x).$$

Ahora basta definir $\langle F\rangle_m$ como la composición de la función $H(x) = x^\frown\langle m\rangle$ con la función G, de modo que

$$\langle F\rangle_m (x) = G(H(x)) = G(x, m).$$

Así, $\ell(\langle F\rangle_m(x)) = m$ y $\langle F\rangle_m (x)_i = F_i(x)$, como se requiere.

Concluimos que una función $F : \mathbb{N}^n \longrightarrow \mathbb{N}^m$ determina, y está determinada por, m funciones $F_i : \mathbb{N}^n \longrightarrow \mathbb{N}$. En estos términos podemos enunciar el teorema siguiente:

Teorema 8.1 (Composición) *Si G cumple que $\bigwedge i < m\ G_i : \mathbb{N}^n \longrightarrow \mathbb{N}$ y $H : \mathbb{N}^m \longrightarrow \mathbb{N}$, existe una única función $F : \mathbb{N}^n \longrightarrow \mathbb{N}$ tal que*

$$\bigwedge x \in \mathbb{N}^n\ F(x) = H(G_0(x), \ldots, G_{m-1}(x)).$$

Basta tomar $F = \langle G\rangle_m \circ H$. Diremos que F es *la función definida por composición* a partir de H y las funciones G_i.

[1] Aquí estamos identificando el conjuno $\mathbb{N}^{n+1}$ con el conjunto $\mathbb{N}^n \times \mathbb{N}$ que obtendríamos al aplicar 7.16.

El teorema siguiente es un caso particular de 7.16 (véase también la nota posterior):

Teorema 8.2 (Recursión) *Si* $G : \mathbb{N}^m \longrightarrow \mathbb{N}$ *y* $H : \mathbb{N}^{m+2} \longrightarrow \mathbb{N}$*, existe una única función* $F : \mathbb{N}^{m+1} \longrightarrow \mathbb{N}$ *tal que*

$$\bigwedge x \in \mathbb{N}^m\, F(x,0) = G(x), \quad \bigwedge x \in \mathbb{N}^m \bigwedge n \in \mathbb{N}\, F(x,n+1) = H(x,n,F(x,n)).$$

Análogamente, si $H : \mathbb{N}^2 \longrightarrow \mathbb{N}$ *y* $a \in \mathbb{N}$*, existe una única* $F : \mathbb{N} \longrightarrow \mathbb{N}$ *tal que*

$$F(0) = a, \quad \bigwedge n \in \mathbb{N}\, F(n+1) = H(n, F(n)).$$

Diremos que F es *la función definida por recursión* a partir de G (o a) y H.

Similarmente, el teorema siguiente es un caso particular de 7.18:

Teorema 8.3 (Minimización) *Sea* $G : \mathbb{N}^{m+1} \longrightarrow \mathbb{N}$ *una función tal que*

$$\bigwedge u \in \mathbb{N}^m \bigvee v\, G(u,v) = 0.$$

Entonces existe una única función $F : \mathbb{N}^m \longrightarrow \mathbb{N}$ *tal que* $F(x)$ *es el mínimo número natural* n *que cumple* $G(x,n) = 0$*.*

Diremos que F es *la función definida por minimización* a partir de G y abreviaremos

$$F(x) = \mu v | G(x,v) = 0.$$

Obviamente, en ACR_0 podemos definir la función sucesor $S : \mathbb{N}^1 \longrightarrow \mathbb{N}$ dada por

$$S \equiv \{u \mid \bigvee vw \leq u(u = \langle v, w\rangle_2 \wedge v \in \mathbb{N}^1 \wedge w = v_0 + 1)\},$$

la función nula $C : \mathbb{N}^1 \longrightarrow \mathbb{N}$ dada por

$$C \equiv \{u \mid \bigvee v \leq u(u = \langle v, 0\rangle_2 \wedge v \in \mathbb{N}^1)\},$$

y, para $0 \leq i < k$, las proyecciones $P_i^n : \mathbb{N}^n \longrightarrow \mathbb{N}$ dadas por

$$P_i^n \equiv \{u \mid \bigvee vw \leq u(u = \langle v, w\rangle_2 \wedge v \in \mathbb{N}^n \wedge w = v_i)\},$$

de modo que

$$\bigwedge u\, S(u) = u + 1, \qquad \bigwedge u\, C(u) = 0, \qquad \bigwedge iku(i < k \wedge u \in \mathbb{N}^n \to P_i^n(u) = u_i).$$

A las funciones S, C, P_i^n las llamaremos *funciones recursivas elementales.*

En la introducción definimos informalmente las funciones recursivas primitivas como las funciones que se obtienen de las funciones recursivas elementales mediante composición y recursión. Este concepto lo formalizamos a través del lenguaje de la aritmética recursiva primitiva, y ahora vamos a ver otra formalización equivalente, salvo que vamos a admitir también la minimización entre las operaciones válidas para construir funciones recursivas.

En este punto podríamos dar una definición natural del concepto de función recursiva, pero no sería muy operativa porque involucraría cuantificaciones sobre conjuntos que luego nos impediría razonar por inducción. Es preferible dar una definición más técnica, pero más adecuada a las limitaciones de ACR_0. Para ello veremos que todas las funciones recursivas pueden definirse mediante fórmulas, así que definiremos las fórmulas que definen funciones recursivas y, a partir de ellas, definiremos las funciones recursivas.

Para ello necesitamos algunas consideraciones previas. Recordemos que en 6.13 hemos definido la fórmula $\mathbb{N} \vDash_{\Sigma_1} \alpha[v]$, de tipo Σ_1, que tiene sentido cuando $\alpha \in \mathrm{Form}(\ulcorner\mathcal{L}_a\urcorner)$ es una fórmula de tipo Σ_1 y v es una aplicación (en el sentido aritmético) definida sobre un conjunto de variables de $\ulcorner\mathcal{L}_a\urcorner$ que contiene a las variables libres de α. Vamos a hacerle un pequeño retoque. Para cada s, definimos el conjunto

$$v_s \equiv \{\langle x_0, s_0\rangle_2, \ldots, \langle x_{\ell(s)\dot{-}1}, s_{\ell(s)\dot{-}1}\rangle_2\}$$

en el sentido aritmético (de modo que v_s es un número natural), donde x_i es la variable libre i-ésima de $\ulcorner\mathcal{L}_a\urcorner$, es decir, que v_s es la aplicación que a la variable x_i le hace corresponder el número natural s_i. Es fácil ver que se trata de un término Δ_1 (porque en ARP se puede definir mediante un funtor).

Diremos que α *es una fórmula de* $\ulcorner\mathcal{L}_a\urcorner$ *con* r *variables* si $\alpha \in \mathrm{Form}(\ulcorner\mathcal{L}_a\urcorner)$ y sus variables libres están entre $x_0, \ldots, x_{r\dot{-}1}$. Así, si α es una fórmula con r variables de tipo Σ_1 y $s \in \mathbb{N}^r$, podemos definir

$$\mathbb{N} \vDash_{\Sigma_1} \alpha[s] \equiv \mathbb{N} \vDash_{\Sigma_1} \alpha[v_s],$$

que es una fórmula de $\mathcal{L}_a$ (en particular de $\mathcal{L}_a^2$) de tipo Σ_1, que expresa que la fórmula α es satisfecha cuando la variable x_i se interpreta como s_i.

Diremos que una fórmula α con $r+1$ variables de tipo Σ_1 define una función $F : \mathbb{N}^r \longrightarrow \mathbb{N}$ (o que F está definida por α) si

$$\bigwedge s \in \mathbb{N}^r \bigwedge n(F(s) = n \leftrightarrow \mathbb{N} \vDash_{\Sigma_1} \alpha[s^\frown\langle n\rangle]).$$

El resultado básico es el siguiente:

Teorema 8.4 *Se cumplen los hechos siguientes:*

1. *Las funciones S, C, P_i^n están definidas, respectivamente, por las fórmulas*
$$\ulcorner x_1 = x_0 + 1\urcorner, \qquad \ulcorner x_1 = 0\urcorner, \qquad \ulcorner x_n = x_i\urcorner.$$

2. *Si una función $H : \mathbb{N}^m \longrightarrow \mathbb{N}$ está definida por la fórmula β y las funciones $G_i : \mathbb{N}^n \longrightarrow \mathbb{N}$, para $i = 0, \ldots, m \dot{-} 1$, están definidas por las fórmulas α_i, entonces la fórmula*
$$\bigvee u_0 \cdots u_{m\dot{-}1}(\alpha_0(x_0, \ldots, x_{n\dot{-}1}, u_0) \wedge \cdots \wedge \alpha_{m\dot{-}1}(x_0, \ldots, x_{n\dot{-}1}, u_{m\dot{-}1}) \wedge \beta(u_0, \ldots, u_{m\dot{-}1}, x_n))$$
define la composición $F : \mathbb{N}^n \longrightarrow \mathbb{N}$ de H con las funciones G_i.

3a. Si la función $G : \mathbb{N}^n \longrightarrow \mathbb{N}$ *está definida por una fórmula* α *y la función* $H : \mathbb{N}^{n+2} \longrightarrow \mathbb{N}$ *está definida por* β*, entonces la fórmula*

$$\bigvee s(\ell(s) = x_n + 1 \wedge \alpha(x_0, \ldots, x_{n\dot{-}1}, s_0) \wedge$$

$$\bigwedge i < x_n\, \beta(x_0, \ldots, x_{n\dot{-}1}, i, s_i, s_{i+1}) \wedge x_{n+1} = s_{x_n})$$

define la función $F : \mathbb{N}^{n+1} \longrightarrow \mathbb{N}$ *definida por recursión a partir de* G *y* H*.*

3b. Si la función $H : \mathbb{N}^2 \longrightarrow \mathbb{N}$ *está definida por la fórmula* β*, entonces la fórmula*

$$\bigvee s(\ell(s) = x_0 + 1 \wedge s_0 = 0^{(a)} \wedge \bigwedge i < x_0\, \beta(i, s_i, s_{i+1}) \wedge x_1 = s_{x_0})$$

define la función $F : \mathbb{N}^1 \longrightarrow \mathbb{N}$ *definida por recursión a partir de* a *y* H*.*

4. Si la función $G : \mathbb{N}^{n+1} \longrightarrow \mathbb{N}$ *está definida por la fórmula* α *y define una función* $F : \mathbb{N}^n \longrightarrow \mathbb{N}$ *por minimización, entonces* F *está definida por la fórmula*

$$\bigwedge i < x_n \bigvee u(\alpha(x_0, \ldots, x_{n\dot{-}1}, i, u) \wedge u \neq 0) \wedge \alpha(x_0, \ldots, x_n, 0).$$

Demostración: 1. Basta observar que

$$\begin{aligned} \mathbb{N} \vDash_{\Sigma_1} {}^{\ulcorner}x_1 = x_0 + 1^{\urcorner}[s^\frown\langle m\rangle] &\leftrightarrow m = S(s), \\ \mathbb{N} \vDash_{\Sigma_1} {}^{\ulcorner}x_1 = 0^{\urcorner}[s^\frown\langle m\rangle] &\leftrightarrow m = C(s), \\ \mathbb{N} \vDash_{\Sigma_1} {}^{\ulcorner}x_n = x_i^{\urcorner}[s^\frown\langle m\rangle] &\leftrightarrow m = P_i^n(s). \end{aligned}$$

2. Similarmente:

$$\mathbb{N} \vDash_{\Sigma_1} \bigvee u_0 \cdots u_{m\dot{-}1}(\alpha_0(x_0, \ldots, x_{n\dot{-}1}, u_0) \wedge \cdots$$

$$\ldots \wedge \alpha_{m\dot{-}1}(x_0, \ldots, x_{n\dot{-}1}, u_{m\dot{-}1}) \wedge \beta(u_0, \ldots, u_{m\dot{-}1}, x_n))[s^\frown\langle k\rangle] \leftrightarrow$$

$$\bigvee t(\ell(t) = m \wedge \bigwedge i < m\ G_i(s) = t_i \wedge H(t) = k) \leftrightarrow F(s) = k.$$

3a.

$$\mathbb{N} \vDash_{\Sigma_1} \bigvee t(\ell(t) = x_n + 1 \wedge \alpha(x_0, \ldots, x_{n\dot{-}1}, t_0) \wedge$$

$$\bigwedge i < x_n\, \beta(x_0, \ldots, x_{n\dot{-}1}, i, t_i, t_{i+1}) \wedge x_{n+1} = t_{x_n})[s^\frown\langle u\rangle] \leftrightarrow$$

$$\bigvee t(\ell(t) = s_n + 1 \wedge G(s|_n) = t_0 \wedge \bigwedge i < s_n H(s|_n{}^\frown\langle i, t_i\rangle) = t_{i+1} \wedge u = t_{s_n})$$

$$\leftrightarrow F(s) = u.$$

Para probar la última equivalencia suponemos la fórmula superior y probamos por inducción que

$$\bigwedge i \leq s_n\, t_i = F(s|_n{}^\frown\langle i\rangle),$$

lo que, para $i = s_n$, implica que $F(s) = t_{s_n} = u$. Recíprocamente, si suponemos que $F(s) = u$, basta tomar $t \in \mathbb{N}^{s_n+1}$ tal que $t_i = F(s|_n{}^\frown\langle i\rangle)$ y se comprueba que t cumple lo requerido por la fórmula superior.

El caso 3b. es análogo.

4.
$$\mathbb{N} \vDash_{\Sigma_1} (\bigwedge i < x_n \bigvee m(\alpha(x_0, \ldots, x_{n\dot{-}1}, i, m) \wedge m \neq 0) \wedge \alpha(x_0, \ldots, x_n, 0))[s^\frown\langle u\rangle] \leftrightarrow \bigwedge i < u \bigvee m(G(s^\frown\langle i\rangle) = m \wedge m \neq 0) \wedge G(s^\frown\langle u\rangle) = 0 \leftrightarrow F(s) = u.$$ ∎

A partir de aquí podríamos definir las fórmulas que definen funciones recursivas, pero es más práctico definir códigos que contengan la misma información que las fórmulas, pero sean técnicamente más simples. Luego asociaremos una fórmula a cada código y finalmente una función a cada fórmula de cada código.

Definición 8.5 Un número natural c es un *código (de una función recursiva parcial)*[2] si y sólo si se da uno de los casos siguientes:

1. $c = \langle 1, 1\rangle$ o $c = \langle 2, 1\rangle$ o existen $j < n$ tales que $c = \langle 3, n, j\rangle$.
2. Existen $c', \bar{c}, m, n$ tales que $m = \ell(\bar{c}) = c'_1$, c' es un código, $\bigwedge i < m(\bar{c}_i$ es un código $\wedge\ (\bar{c}_i)_1 = n)$ y $c = \langle 4, n, c', \bar{c}\rangle$.

3a. Existen n y códigos c', c'' tales que $c'_1 = n$, $c''_1 = n{+}2$ y $c = \langle 5, n+1, c', c''\rangle$.

3b. Existen un código c' y a tales que $c'_1 = 2$ y $c = \langle 6, 1, a, c'\rangle$.

4. Existen un código c' y $n \geq 1$ tales que $c'_1 = n + 1$ y $c = \langle 7, n, c'\rangle$.

Notemos que hemos definido cuándo un número natural c es un código suponiendo que ya sabemos si los números menores que c lo son. Esto se formaliza en ACR_0 a través del teorema de recursión completa 7.17, pues con él podemos definir una función $F : \mathbb{N} \longrightarrow \{0, 1\}$ que a su vez nos permite definir los códigos como los números naturales que cumplen $F(c) = 1$. La función F se define en la forma $F(c) = G(c, F|_c)$, para cierta función $G : \mathbb{N} \times \mathbb{N} \longrightarrow \mathbb{N}$ definida por Δ^0_1-comprensión que se corresponde con los apartados de la definición que acabamos de dar. Sólo hay que observar que, en todos ellos, todos los cuantificadores pueden acotarse por c.

A su vez, podemos definir el conjunto $\mathcal{C}$ de todos los códigos, que técnicamente es
$$\mathcal{C} = \{c \mid F(c) = 1\}.$$

Más técnicamente, la definición anterior puede formalizarse en ARP mediante un funtor F definido por el teorema de recursión completa 2.9, lo que nos da una fórmula atómica $c \in \mathcal{C}$ de $\mathcal{L}_{\mathrm{arp}}$ que cumple las mismas condiciones de la definición precedente. Ésta a su vez se traduce a una fórmula Δ_1 de $\mathcal{L}_a$ que a su vez se corresponde con una fórmula Δ_1 (no meramente Δ^0_1, es decir, sin variables de segundo orden) de $\mathcal{L}^2_a$. Así pues, podemos afirmar que la fórmula $c \in \mathcal{C}$ es equivalente en ACR_0 a una fórmula Δ_1.

[2]El concepto de función recursiva parcial lo introduciremos en la sección 8.3. De momento nos referiremos a estos códigos simplemente como "códigos", y en dicha sección veremos su relación con las funciones recursivas parciales.

Si $c \in \mathcal{C}$, en cualquiera de los casos de la definición se cumple que $\ell(c) \geq 2$. Al número c_0 lo llamaremos *tipo* del código (y hay siete tipos posibles) y al número $\mathrm{Nar}(c) \equiv c_1$ lo llamaremos *número de argumentos* del código.

También son claros los hechos siguientes:

1a. Hay un único código de tipo 1, que es $\hat{S} \equiv \langle 1, 1\rangle$, de modo que $\mathrm{Nar}(\hat{S}) = 1$.

1b. Hay un único código de tipo 2, que es $\hat{C} \equiv \langle 2, 1\rangle$, y también $\mathrm{Nar}(\hat{C}) = 1$.

1c. Los códigos de tipo 3 son los de la forma $\hat{P}^n_j \equiv \langle 3, n, j\rangle$, con $j < n$, de modo que $\mathrm{Nar}(\hat{P}^n_j) = n$.

2. Los códigos de tipo 4 son los de la forma

$$\kappa(c, \langle c_1, \ldots, c_m\rangle) \equiv \langle 4, n, c, \langle c_1, \ldots, c_m\rangle\rangle\,,$$

donde c es un código de m argumentos y cada c_i es un código de n argumentos, y entonces $\mathrm{Nar}(\kappa(c, \langle c_1, \ldots, c_m\rangle)) = n$.

3a. Los códigos de tipo 5 son los de la forma $\rho(c, c') \equiv \langle 5, n+1, c, c'\rangle$, donde c es un código de n argumentos y c' un código de $n+2$ argumentos, y entonces $\mathrm{Nar}(\rho(c, c')) = n + 1$.

3b. Los códigos de tipo 6 son los de la forma $\rho^*(a, c) \equiv \langle 6, 1, a, c\rangle$, donde c es un código de 2 argumentos, y entonces $\mathrm{Nar}(\rho^*(a, c)) = 1$.

4. Los códigos de tipo 7 son los de la forma $\mu(c) \equiv \langle 7, n, c\rangle$, donde c es un código con $n + 1$ argumentos, y entonces $\mathrm{Nar}(\mu(c)) = n$.

Estos códigos pueden verse como una versión simplificada de los funtores de $\mathcal{L}_{\mathrm{arp}}$, salvo por el hecho de que hemos incluido códigos que determinen funciones definidas por minimización.

Si en la definición de código eliminamos el punto 4. obtenemos la definición de un conjunto $\mathcal{C}_{\mathrm{rp}} \subset \mathcal{C}$ tal que la fórmula $c \in \mathcal{C}_{\mathrm{rp}}$ es también equivalente a una fórmula de tipo Δ_1 (no meramente Δ^0_1).

Ahora usamos el teorema de recursión completa 7.17 para definir una función $\mathcal{F} : \mathbb{N} \longrightarrow \mathrm{Form}(\ulcorner \mathcal{L}_a \urcorner)$ según los criterios siguientes:

1a. Si $c = \hat{S}$, entonces $\mathcal{F}(c) = \ulcorner x_1 = x_0 + 1\urcorner$.

1b. Si $c \notin \mathcal{C}$ o $c = \hat{C}$, entonces $\mathcal{F}(c) = \ulcorner x_1 = 0\urcorner$.

1c. Si $c = \hat{P}^n_j$, con $j < n$, entonces $\mathcal{F}(c) = \ulcorner x_n = x_j\urcorner$.

2. Si $c = \kappa(c', \langle c_1, \ldots, c_m\rangle)$, $\mathrm{Nar}(c_i) = n$ y $\mathcal{F}(c') = \beta$, $\mathcal{F}(c_i) = \alpha_i$, entonces

$$\mathcal{F}(c) = \bigvee u_0 \cdots u_{m\dot{-}1}(\alpha_0(x_0, \ldots, x_{n\dot{-}1}, u_0) \wedge \cdots \wedge \alpha_{m\dot{-}1}(x_0, \ldots, x_{n\dot{-}1}, u_{m\dot{-}1})$$
$$\wedge\, \beta(u_0, \ldots, u_{m\dot{-}1}, x_n)).$$

3a. Si $c = \rho(c', c'')$, con $\mathrm{Nar}(c') = n$, $\mathcal{F}(c') = \alpha$, $\mathcal{F}(c'') = \beta$, entonces

$$\mathcal{F}(c) = \bigvee s(\ell(s) = x_n + 1 \wedge \alpha(x_0, \ldots, x_{n \dot{-} 1}, s_0) \wedge$$

$$\bigwedge i < x_n \, \beta(x_0, \ldots, x_{n \dot{-} 1}, i, s_i, s_{i+1}) \wedge x_{n+1} = s_{x_n}).$$

3b. Si $c = \rho^*(a, c')$ y $\mathcal{F}(c') = \beta$, entonces

$$\bigvee s(\ell(s) = x_0 + 1 \wedge s_0 = 0^{(a)} \wedge \bigwedge i < x_0 \, \beta(i, s_i, s_{i+1}) \wedge x_1 = s_{x_0}).$$

4. Si $c = \mu(c')$ y $\mathrm{Nar}(c') = n + 1$, $\mathcal{F}(c') = \alpha$, entonces

$$\mathcal{F}(c) = \bigwedge i < x_n \bigvee u(\alpha(x_0, \ldots, x_{n \dot{-} 1}, i, u) \wedge u \neq 0) \wedge \alpha(x_0, \ldots, x_n, 0).$$

Nuevamente, la función $\mathcal{F}$ se puede definir (como un funtor) en ARP, y esto se traduce en que la fórmula $\mathcal{F}(c) = \alpha$ es de tipo Δ_1 en $\mathcal{L}_a^2$ (no meramente Δ_1^0).

Una simple inducción muestra que, si $c \in \mathcal{C}$ cumple $\mathrm{Nar}(c) = n$, entonces $\mathcal{F}(c)$ es una fórmula de $\ulcorner \mathcal{L}_a \urcorner$ de tipo Σ_1 con a lo sumo las variables libres $x_0, \ldots, x_n$.

Por el convenio que hemos adoptado en el punto 1b., esto es válido también si $c \notin \mathcal{C}$ si convenimos en que, en este caso, $\mathrm{Nar}(c) = 1$.

Finalmente definimos la fórmula Σ_1 de $\mathcal{L}_a$ dada por

$$\{c\}(s) = n \equiv \mathbb{N} \vDash_{\Sigma_1} \mathcal{F}(c)[s^\frown\langle n\rangle].$$

Aquí hay que entender que $\{c\}(s) = n$ no es una igualdad, sino una fórmula de $\mathcal{L}_a$ con tres variables libres: c, s, n. Ahora bien, la notación como igualdad está parcialmente justificada por el teorema siguiente:

Teorema 8.6 $\bigwedge csuv(\ell(s) = \mathrm{Nar}(c) \wedge \{c\}(s) = u \wedge \{c\}(s) = v \to u = v)$.

Demostración: Probamos por inducción completa sobre c la fórmula Π_1

$$\bigwedge suv(\ell(s) = \mathrm{Nar}(c) \wedge \{c\}(s) = u \wedge \{c\}(s) = v \to u = v).$$

Para ello distinguimos casos según el tipo de c.

Si $c = \hat{S}$, entonces $\ell(s) = 1$ y

$$\{c\}(s) = u \leftrightarrow \mathbb{N} \vDash_{\Sigma_1} \ulcorner x_1 = x_0 + 1 \urcorner[s^\frown\langle u\rangle] \leftrightarrow u = S(s),$$

luego, ciertamente, si $\{c\}(s) = u \wedge \{c\}(s) = v$, tenemos que $u = S(s) = v$.

Si $c = \hat{C}$ (o también si $c \notin \mathcal{C}$), tenemos que $\ell(s) = 1$ y

$$\{c\}(s) = u \leftrightarrow \mathbb{N} \vDash_{\Sigma_1} \ulcorner x_1 = 0 \urcorner[s^\frown\langle u\rangle] \leftrightarrow u = C(s),$$

y llegamos a la misma conclusión.

Si $c = \hat{P}^n_j$, tenemos que $\ell(s) = n$ y

$$\{c\}(s) = u \leftrightarrow \mathbb{N} \vDash_{\Sigma_1} \ulcorner x_u = x_j \urcorner[s^\frown\langle u\rangle] \leftrightarrow u = P^n_j(s),$$

y llegamos a la misma conclusión.

Si $c = \kappa(c', \langle c_1, \ldots, c_m\rangle)$, donde $\mathrm{Nar}(c_i) = n$, entonces $\ell(s) = n$ y, llamando $\alpha_i = \mathcal{F}(c_i)$, $\beta = \mathcal{F}(c')$, como hemos visto en la prueba del teorema 8.4, tenemos

$$\{c\}(s) = u \leftrightarrow \bigvee t(\ell(t) = m \wedge \bigwedge i < m\{c_i\}(s) = t_i \wedge \{c'\}(t) = u).$$

Por lo tanto, si también se cumple $\{c\}(s) = v$, tenemos que

$$\bigvee t'(\ell(t') = m \wedge \bigwedge i < m\{c_i\}(s) = t'_i \wedge \{c'\}(t') = v),$$

pero, por hipótesis de inducción, $\bigwedge i < m\, t_i = t'_i$, luego $t = t'$, luego (de nuevo por hipótesis de inducción) $u = v$.

Si $c = \rho(c', c'')$, donde $\mathrm{Nar}(c') = n$, $\mathcal{F}(c') = \alpha$, $\mathcal{F}(c'') = \beta$, entonces tenemos que $\ell(s) = n + 1$ y, según hemos visto en la prueba de 8.4,

$$\{c\}(s) = u \leftrightarrow \bigvee t(\ell(t) = s_n + 1 \wedge \{c'\}(s|_n) = t_0 \wedge$$

$$\bigwedge i < s_n\{c''\}(s|_n{}^\frown\langle i, t_i\rangle) = t_{i+1} \wedge u = t_{s_n}).$$

Si se cumple también $\{c\}(s) = v$, tenemos que

$$\bigvee t'(\ell(t') = s_n + 1 \wedge \{c'\}(s|_n) = t'_0 \wedge \bigwedge i < s_n\{c''\}(s|_n{}^\frown\langle i, t'_i\rangle) = t'_{i+1} \wedge v = t'_{s_n}),$$

luego, por hipótesis de inducción (para c'), tenemos que $t_0 = t'_0$ y por inducción se prueba que $\bigwedge i \le s_n\, t_i = t'_i$, pues lo tenemos probado para $i = 0$ y, si vale para i, entonces, como

$$\{c''\}(s|_n{}^\frown\langle i, t_i\rangle) = t_{i+1}, \qquad \{c''\}(s|_n{}^\frown\langle i, t_i\rangle) = t'_{i+1},$$

por hipótesis de inducción es $t_{i+1} = t'_{i+1}$, luego $u = t_{s_n} = t'_{s_n} = v$.

El caso $c = \rho^*(a, c')$ se razona de forma análoga.

Si $c = \mu(c')$, con $\mathrm{Nar}(c') = n + 1$, $\mathcal{F}(c') = \alpha$, entonces $\ell(s) = n$ y

$$\{c\}(s) = u \leftrightarrow \bigwedge i < u \bigvee m(\{c'\}(s^\frown\langle i\rangle) = m \wedge m \neq 0) \wedge \{c'\}(s^\frown\langle u\rangle) = 0.$$

Si también se cumple $\{c\}(s) = v$, entonces

$$\bigwedge i < v \bigvee m(\{c'\}(s^\frown\langle i\rangle) = m \wedge m \neq 0) \wedge \{c'\}(s^\frown\langle v\rangle) = 0,$$

pero tiene que ser $u = v$ pues, si fuera, por ejemplo, $u < v$, de $\{c\}(s) = v$ se sigue que existe un $m \neq 0$ tal que $\{c'\}(s^\frown\langle u\rangle) = m$, mientras que de $\{c\}(s) = u$ se sigue que $\{c'\}(s^\frown\langle u\rangle) = 0$, y la hipótesis de inducción nos da que $m = 0$. ∎

No obstante, la fórmula $\{c\}(s) = n$ no se comporta exactamente como una igualdad porque, dado un s, puede que no haya ningún n que la satisfaga:

Definición 8.7 Usaremos la notación

$$\{c\}(s)\!\downarrow \equiv \bigvee n\,\{c\}(s) = n, \qquad \{c\}(s)\!\uparrow \equiv \neg\bigvee n\,\{c\}(s) = n.$$

Si pensamos en $\{c\}$ como "la función definida por el código c" (que todavía no hemos definido), entonces $\{c\}(s)\!\downarrow$ significa que $\{c\}$ está definida en s, mientras que $\{c\}(s)\!\uparrow$ significa que $\{c\}$ no está definida en s.

Diremos que un código $c \in \mathcal{C}$ (resp. $c \in \mathcal{C}_{\mathrm{rp}}$) de r argumentos es *un código de una función recursiva (primitiva)* si $\bigwedge s \in \mathbb{N}^r\,\{c\}(s)\!\downarrow$.

Observemos que la fórmula $\{c\}(s)\!\downarrow$ es Σ_1, mientras que "c es un código de una función recursiva" es Π_2.

Si c es el código de una función recursiva y $\mathrm{Nar}(c) = r$, entonces

$$\bigwedge s \in \mathbb{N}^r(\{c\}(s) = n \leftrightarrow \bigwedge m(\{c\}(s) = m \to m = n)).$$

En efecto, la implicación $\to$ es consecuencia del teorema anterior, mientras que, si se cumple la parte de la derecha, tenemos que, fijado $s \in \mathbb{N}^r$, existe un m tal que $\{c\}(s) = m$, luego $m = n$, luego $\{c\}(s) = n$.

Equivalentemente,

$$\ell(s) = \mathrm{Nar}(c) \wedge \{c\}(s) = n \leftrightarrow \ell(s) = \mathrm{Nar}(c) \wedge \bigwedge m(\{c\}(s) = m \to m = n),$$

donde la primera fórmula es Σ_1 y la segunda es Π_1, por lo que podemos usar el esquema de Δ_1^0-comprensión para definir el conjunto

$$\{c\} = \{z \mid \bigvee sn \le z(z = \langle s, n\rangle_2 \wedge \ell(s) = \mathrm{Nar}(c) \wedge \{c\}(s) = n)\},$$

de modo que $\{c\} : \mathbb{N}^r \longrightarrow \mathbb{N}$.

Diremsos que una función $F : \mathbb{N}^r \longrightarrow \mathbb{N}$ es *recursiva (primitiva)* si existe un código c para una función recursiva (primitiva) tal que $F = \{c\}$.

El teorema siguiente resume lo que hemos obtenido:

Teorema 8.8 *Si c es un código para una función recursiva (primitiva) de modo que $\mathrm{Nar}(c) = r$, entonces $\{c\} : \mathbb{N}^r \longrightarrow \mathbb{N}$ es una función recursiva (primitiva). Además, para todo $t \in \mathbb{N}^{r+1}$,*

$$t \in G(\{c\}) \leftrightarrow \{c\}(t|_r) = t_r \leftrightarrow \bigwedge m(\{c\}(t|_r) = m \to m = t_r),$$

donde la fórmula central es Σ_1 y la última es Π_1.

Más aún, el teorema 8.4 nos da inmediatamente el teorema siguiente:

Teorema 8.9 *Se cumple:*

1. *Los códigos $\hat{S}, \hat{C}, \hat{P}_j^n$ son códigos de funciones recursivas primitivas, y se cumple que $\{\hat{S}\} = S$, $\{\hat{C}\} = C$, $\{\hat{P}_j^n\} = P_j^n$.*
2. *Si $c_1, \ldots, c_m$ son códigos de funciones n-ádicas recursivas (primitivas) y c' es un código de una función m-ádica recursiva (primitiva), entonces $c = \kappa(c', \langle c_1, \ldots, c_m\rangle)$ es un código de una función n-ádica recursiva (primitiva), y $\{c\} : \mathbb{N}^n \longrightarrow \mathbb{N}$ es la función definida por composición a partir de $\{c'\} : \mathbb{N}^m \longrightarrow \mathbb{N}$ y de las funciones $\{c_i\} : \mathbb{N}^n \longrightarrow \mathbb{N}$.*

3a. Si c' es un código de una función n-ádica recursiva (primitiva) y c'' es un código de una función $n+2$-ádica recursiva (primitiva), entonces $c = \rho(c', c'')$ es un código de una función $n+1$-ádica recursiva (primitiva), y $\{c\} : \mathbb{N}^{n+1} \longrightarrow \mathbb{N}$ es la función definida por recursión a partir de las funciones $\{c'\} : \mathbb{N}^n \longrightarrow \mathbb{N}$ y $\{c''\} : \mathbb{N}^{n+2} \longrightarrow \mathbb{N}$.

3b Si c' es un código de una función diádica recursiva (primitiva), entonces $c = \rho^*(a, c')$ es el código de una función monádica recursiva (primitiva), y $\{c\} : \mathbb{N} \longrightarrow \mathbb{N}$ es la función definida por recursión a partir de a y de la función $\{c'\} : \mathbb{N}^2 \longrightarrow \mathbb{N}$.

4. Si c' es el código de una función $n+1$-ádica recursiva tal que

$$\bigwedge u \in \mathbb{N}^n \bigvee v \, \{c'\}(u) = v,$$

entonces $c = \mu(c')$ es el código de una función n-ádica recursiva, y la función $\{c\} : \mathbb{N}^n \longrightarrow \mathbb{N}$ es la definida por minimización a partir de la función $\{c'\} : \mathbb{N}^{n+1} \longrightarrow \mathbb{N}$.

Sucede que, aunque c' sea un código de una función recursiva, si no cumple la condición exigida por el punto 4. del teorema anterior, puede suceder que el código $c = \mu(c')$ no sea un código de una función recursiva, pues puede ocurrir que $\{c\}(s)\!\uparrow$ para algunos valores de s. Éste es el único inconveniente que puede hacer que un código c no defina una función recursiva, pero no podemos probarlo en ACR_0, sino que necesitamos suponer el principio de Σ_2-inducción:

Teorema 8.10 ($\boldsymbol{\Sigma_2}$-inducción) *Todo código $c \in \mathcal{C}_{\mathrm{rp}}$ define una función recursiva primitiva.*

Demostración: Esto es inmediato a partir del teorema anterior, pues nos permite probar, por inducción sobre c, que

$$c \in \mathcal{C}_{\mathrm{rp}} \to \bigwedge s(\ell(s) = \mathrm{Nar}(c) \to \bigvee n\{c\}(s) = n),$$

pero, como la fórmula es de tipo Π_2, necesitamos el principio de Π_2-inducción (que es equivalente al de Σ_2-inducción). ∎

Así pues, las funciones $\{c\}$, con $c \in \mathcal{C}_{\mathrm{rp}}$ son una forma de formalizar en ACR_0 las funciones recursivas primitivas[3] que resulta ser una alternativa a los funtores de $\mathcal{L}_{\mathrm{arp}}$, con la diferencia de que esta formalización nos permite formular enunciados generales de la forma "para toda función recursiva primitiva" o "existe una función recursiva primitiva", mientras que en ARP no podemos decir "para todo funtor", sino que una afirmación de este tipo es necesariamente metamatemática.

La situación es que en ACR_0 podemos probar una versión equivalente de cualquier resultado que podamos probar en ARP sobre funciones recursivas primitivas, y a menudo generalizarlo a funciones recursivas, pero a la hora de

[3]Teniendo en cuenta que la fórmula $\{c\}(s) = n$ es Σ_1, de modo que puede expresarse sin variables de segundo orden, en realidad podríamos haber desarrollado toda la teoría en $\mathrm{I}\Sigma_1$. No obstante, al trabajar en ACR_0 simplificamos los enunciados relativos a funciones.

formular enunciados generales no expresables en $\mathcal{L}_{\text{arp}}$ podemos necesitar principios más fuertes, como ilustra el teorema anterior, donde la necesidad de la Σ_2-inducción está relacionada con la imposibilidad de demostrar en ARP el falso teorema de la página 126.

Por ejemplo, ahora es evidente que si vamos definiendo funciones por composición, recursión o minimización a partir de las funciones S, C, P_j^n, las funciones obtenidas son recursivas (y, de hecho, recursivas primitivas si no usamos la minimización), pues podemos ir definiendo sus códigos correspondientes mediante el teorema 8.9, el cual nos garantiza que tales códigos definen las funciones consideradas.

En particular, cada definición de un funtor en ARP da lugar a un código (a un numeral concreto) de una función recursiva primitiva en ACR_0 y a la función recursiva primitiva correspondiente, por lo que podemos dar por demostrado que todas las funciones expresables mediante funtores de $\mathcal{L}_{\text{arp}}$ son recursivas primitivas, como es el caso de $x + y$, $x \dot{-} y$, $\text{máx}\{x, y\}$, etc.

Por ejemplo, el hecho de que el teorema 2.9 sea demostrable en ARP se traduce en la versión siguiente en ACR_0:

Teorema 8.11 (Recursión completa) *Si $G : \mathbb{N}^{m+2} \longrightarrow \mathbb{N}$ es recursiva (primitiva), la función $F : \mathbb{N}^{m+1} \longrightarrow \mathbb{N}$ dada por*

$$\bigwedge u \in \mathbb{N}^m \bigwedge n \in \mathbb{N}\, F(x, n) = G(x, n, F|_n(x))$$

es recursiva (primitiva).

En principio lo podemos considerar demostrado para funciones recursivas primitivas, pero la prueba del teorema 7.17 muestra que si G es recursiva, la función H también lo es, y a su vez esto implica que F también lo es.

Notemos que el apartado 3b del teorema 8.9 no se corresponde con ninguna definición de un funtor en ARP, pero sí con el apartado 3 del teorema 1.15. Esto se traduce en que podríamos haber eliminado el apartado 3b de la definición de código sin alterar por ello el concepto de "función recursiva", pues el carácter recursivo (o recursivo primitivo) de las funciones de tipo 3b podría demostrarse a partir de la definición reducida.

Conviene observar también que la minimización acotada define funciones recursivas primitivas:

Teorema 8.12 (Minimización acotada) *Si $G : \mathbb{N}^{m+1} \longrightarrow \mathbb{N}$ es recursiva (primitiva), existe una única función $F : \mathbb{N}^{m+1} \longrightarrow \mathbb{N}$ que cumple*

$$F(x, y) = \mu u \leq y\, G(x, u) = 0$$

y es recursiva (primitiva), donde hay que entender que $F(x, y) = 0$ si no existe ningún $u \leq y$ que cumpla $G(x, u) = 0$.

DEMOSTRACIÓN: Definimos $H : \mathbb{N}^{m+1} \longrightarrow \mathbb{N}$ mediante

$$H(x,0) = 1 \dot{-} (1 \dot{-} G(x,0)), \quad H(x,y+1) = H(x,y) \cdot (1 \dot{-} (1 \dot{-} G(x,y+1))).$$

Es claro que

$$H(x,y) = \begin{cases} 0 & \text{si existe } u \leq y \text{ tal que } G(x,y) = 0, \\ 1 & \text{en caso contrario.} \end{cases}$$

Ahora definimos F por recursión:

$$F(x,0) = 0, \quad F(x,y+1) = F(x,y) + (y+1)(H(x,y) \dot{-} H(x,y+1)).$$

Es fácil ver que F cumple lo requerido. ∎

Un poco más en general, el teorema anterior vale igualmente para definiciones de la forma

$$F(x,y) = \mu u \leq y\, G_1(x,u) = G_2(x,u),$$

pues esto equivale a

$$F(x,y) = \mu u \leq y\, |G_1(x,u) - G_2(x,u)| = 0.$$

En ARP podemos construir fácilmente funtores que tomen cualquier valor constante, pero el enunciado "para todo número natural n, existe un funtor que toma de forma constante el valor n" es necesariamente un metateorema. En este contexto podemos formularlo como un teorema, pero necesitamos nuevamente el principio de Σ_2-inducción:

Teorema 8.13 ($\boldsymbol{\Sigma_2}$-inducción) *Las funciones constantes son recursivas primitivas.*

DEMOSTRACIÓN: Definimos por recursión

$$c_0 = \hat{C}, \qquad c_{n+1} = \kappa(\hat{S}, \langle c_n \rangle),$$

con lo que técnicamente estamos definiendo un código $\hat{c} \in \mathcal{C}_{\mathrm{rp}}$ que, por 8.9, define una función recursiva primitiva $\{\hat{c}\}$, y escribimos $c_n = \{\hat{c}\}(\langle n \rangle)$.

Ahora una simple inducción prueba que $\bigwedge n\, c_n \in \mathcal{C}_{\mathrm{rp}}$ Notemos que la fórmula es Σ_1, pues equivale a

$$\bigvee c(\{\hat{c}\}(\langle n \rangle) = c \wedge c \in \mathcal{C}_{\mathrm{rp}}).$$

A su vez, por inducción sobre n probamos que $\bigwedge nm\, \{c_n\}(\langle m \rangle) = n$, y en este caso la fórmula equivale a

$$\bigwedge mc(\{\hat{c}\}(\langle n \rangle) = c \to \{c\}(\langle m \rangle) = n),$$

que es de tipo Π_2, pues el consecuente[4] de la implicación es Σ_1, mientras que el antecedente es Δ_1, luego lo podemos tomar Π_1, de modo que la implicación es Σ_1, y con los cuantificadores iniciales es Π_2.

[4] Alternativamente, podemos usar el teorema 8.10 para justificar que el consecuente es también Π_1, con lo que, tomando el antecedente Σ_1, la fórmula completa es Π_1, pero igualmente necesitamos el principio de Σ_2-inducción, en este caso a través del uso de 8.10.

Esto prueba que $\{c_n\} : \mathbb{N}^1 \longrightarrow \mathbb{N}$ es la función constante n, que es, pues, recursiva primitiva. Si llamamos $F(r,n) = \kappa(\hat{P}_0^r, \langle c_n\rangle)$ tenemos que, para cada $r \geq 1$, se cumple que $\{F(r,n)\} : \mathbb{N}^r \longrightarrow \mathbb{N}$ es la composición de $P_0^r : \mathbb{N}^r \longrightarrow \mathbb{N}$ con la función constante n, luego es la función constante n de r variables, que también es recursiva primitiva. ∎

Conviene resaltar que en ACR_0 podemos probar que $\{F(r,0)\}$ es la función constante 0 de r variables, y que $\{F(r,1)\}$ es la función constante 1 de r variables, etc., y así podemos probar que cada función constante es recursiva primitiva. La Σ_2-inducción sólo hace falta para reducir este esquema teoremático a un único teorema.

Consideramos ahora un caso muy sutil:

Teorema 8.14 ($\boldsymbol{\Sigma_2}$-inducción) *Para cada $r \geq 1$, la función $F_r : \mathbb{N}^r \longrightarrow \mathbb{N}$ dada por $F_r(s) = s$ es recursiva primitiva.*

Aunque es cierto, esto no es trivial. Aquí es fundamental tener en cuenta el convenio que estamos adoptando para codificar las funciones r-ádicas. Por ejemplo, se cumple que $\langle 2,3\rangle = 173$, pero cuando decimos que la suma es una función recursiva primitiva, queremos decir que lo es la función que toma como argumentos 2 y 3 (en ese orden) y nos devuelve un 5, no la función a la que le damos un 173 y nos devuelve un 5. El 173 es aquí un mero convenio que usamos en las teorías aritméticas para representar el par formado por los números naturales 2 y 3.

Similarmente, lo que tenemos que probar (y no es inmediato) es que la función F_2 a la que si le damos un 2 y un 3 (en ese orden) nos devuelve un 173 es recursiva primitiva. Por la forma en que hemos codificado los pares ordenados, darle un 2 y un 3 a F_2 es técnicamente darle un 173, pero lo que tenemos que probar es que el cálculo que a partir de un 2 y un 3 nos devuelve un 173 corresponde a una función recursiva primitiva.

Al margen de la sutileza conceptual, si el lector se para a pensar lo que hay que demostrar formalmente, verá que no es inmediato. La función "identidad", a la que le damos s y nos devuelve s es técnicamente la función $F : \mathbb{N}^1 \longrightarrow \mathbb{N}$ dada por $F(\langle s\rangle) = s$, que es trivialmente recursiva primitiva, pues no es sino la proyección P_1^1. Pero no es eso lo que tenemos que probar.

Demostración: Consideremos en primer lugar el caso $r = 1$. La función $F_1 : \mathbb{N}^1 \longrightarrow \mathbb{N}$ es la dada por

$$F(\langle x\rangle) = \langle x\rangle = \langle 0, x\rangle_2 = x(x+1)/2,$$

que ciertamente es recursiva primitiva, pero no porque lo sea "la identidad", sino porque lo son la suma, el producto y el cociente euclídeo. Usando las construcciones de estas funciones a partir de las funciones elementales podemos calcular explícitamente un número natural (un numeral) $\hat{F}_1$ para el que, mediante aplicaciones sucesivas del teorema 8.9, podemos probar que $\hat{F}_1 \in \mathfrak{C}_{\mathrm{rp}}$ y que, para todo x,

$$\{\hat{F}_1\}(\langle x\rangle) = x(x+1)/2 = \langle x\rangle\,,$$

por lo que $F_1 = \{\hat{F}_1\}$ es recursiva primitiva.

Supongamos ahora que tenemos un código c para una función recursiva primitiva tal que $F_r = \{c\}$ y vamos a ver cómo podemos construir un código que determine a F_{r+1}.

Si componemos las funciones P_i^{r+1}, para $i < r$ con $\{c\}$, obtenemos la función $G : \mathbb{N}^{r+1} \longrightarrow \mathbb{N}$ dada por $G(s) = s|_r$, cuyo código es $\kappa(c, \langle \hat{P}_0^{r+1}, \ldots, \hat{P}_{r\dot{-}1}^{r+1} \rangle)$.

Ahora componemos esta función y P_r^{r+1} con la función $H : \mathbb{N}^2 \longrightarrow \mathbb{N}$ dada por $H(u, v) = u^\frown\langle v \rangle$, que es recursiva primitiva, y obtenemos la función recursiva primitiva $F : \mathbb{N}^{r+1} \longrightarrow \mathbb{N}$ dada por

$$F(s) = H(s|_r, s_r) = s|_r{}^\frown\langle s_r \rangle = s.$$

Así pues, $F = F_r$. Si llamamos $\hat{H}$ al código de H (que es un numeral que podemos calcular explícitamente), el código de F_r es

$$\bar{c} \equiv \kappa(\hat{H}, \langle \kappa(c, \langle \hat{P}_0^{r+1}, \ldots, \hat{P}_{r\dot{-}1}^{r+1} \rangle), \hat{P}_r^{r+1} \rangle),$$

y hemos probado que $\{\bar{c}\} = F_{r+1}$.

Ahora bien, la función $c \mapsto \bar{c}$ es claramente recursiva primitiva (y podríamos calcular explícitamente un código que la determine), y a su vez, con ella podemos definir la función recursiva primitiva dada por

$$F(0) = \hat{F}_1, \qquad F(r+1) = \overline{F(r)}$$

Para evitar el desfase debido a que $F(0)$ es un código para F_1, definimos la función $c_r = F(r \dot{-} 1)$, que es también una función recursiva primitiva que cumple

$$c_0 = c_1 = \hat{F}_1, \qquad c_{r+1} = \bar{c}_r.$$

Esta función tiene un código $\hat{c}$ que podríamos calcular explícitamente y que nos permite formular la inducción necesaria para demostrar el teorema:

$$r \geq 1 \to \bigwedge s(s \in \mathbb{N}^r \to \{\{\hat{c}\}(r)\}(s) = s)$$

o, más detalladamente:

$$r \geq 1 \to \bigwedge sc(\ell(s) = r \wedge \{\hat{c}\}(\langle r \rangle) = c \to \{c\}(s) = s).$$

El consecuente (de la segunda implicación) es Σ_1 y el antecedente es Δ_1 (porque $\hat{c}$ es el código de una función recursiva primitiva), luego la fórmula es Π_2 y podemos razonar por inducción sobre r. El caso $r = 1$ ya está probado y el hecho de que si se cumple para r también se cumple para $r + 1$ lo hemos probado para motivar la definición de $\bar{c}$. ∎

Nuevamente insistimos en que en ACR_0 hemos definido una función recursiva primitiva que nos da una sucesión de códigos c_r de modo que podemos probar que $F_1 = \{c_1\}$, $F_2 = \{c_2\}$, etc., y sólo necesitamos el principio de Σ_2-inducción para reducir este esquema teoremático a un único teorema.

Funciones recursivas primitivas y funtores de ARP Vamos ahora a explicitar la relación entre las funciones recursivas primitivas y los funtores de la aritmética recursiva primitiva:

Teorema 8.15 ($\boldsymbol{\Sigma_2}$-inducción) *Si f es un funtor r-ádico de $\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner$, la traducción a $\ulcorner\mathcal{L}_a\urcorner$ de la fórmula $f(x_0,\dots,x_{r\dot{-}1})=x_r$ según el teorema 4.49 define una función recursiva primitiva $F:\mathbb{N}^r\longrightarrow\mathbb{N}$.*

Demostración: Observemos que en ARP se demuestra obviamente que

$$\bigwedge x_0\cdots x_{r\dot{-}1}\overset{1}{\bigvee}x_r\, f(x_0,\dots,x_{r\dot{-}1})=x_r,$$

luego en $\mathrm{I}\Sigma_1$ se demuestra la traducción a $\ulcorner\mathcal{L}_a\urcorner$ de esta fórmula, luego, según el teorema 6.15, también se demuestra

$$\mathbb{N}\vDash_{\Sigma_1}\bigwedge x_0\cdots x_{r\dot{-}1}\overset{1}{\bigvee}x_r\, f(x_0,\dots,x_{r\dot{-}1})=x_r,$$

y esto equivale a que

$$\bigwedge s\in\mathbb{N}^r\overset{1}{\bigvee}n\ \mathbb{N}\vDash_{\Sigma_1}(f(x_0,\dots,x_{r\dot{-}1})=x_r)[s,n].$$

Veamos ahora que podemos definir

$$F_f\equiv\{z\mid\bigvee sn\le z(z=\langle s,n\rangle_2\wedge\ell(s)=r\wedge\mathbb{N}\vDash_{\Sigma_1}(f(x_0,\dots,x_{r\dot{-}1})=x_r)[s,n])\}.$$

Para aplicar el principio de Δ^0_1-comprensión tenemos que probar que la fórmula que define a F_f, que obviamente es Σ_1, equivale a una fórmula Π_1 y, en efecto, equivale a

$$\bigvee sn\le z(z=\langle s,n\rangle_2\wedge\ell(s)=r\wedge$$

$$\bigwedge m(\mathbb{N}\vDash_{\Sigma_1}(f(x_0,\dots,x_{r\dot{-}1})=x_r)[s,m]\to m=n)).$$

Es claro entonces que $F_f:\mathbb{N}^r\longrightarrow\mathbb{N}$ y

$$\bigwedge s\in\mathbb{N}^r\bigwedge n(F_f(s)=n\leftrightarrow\mathbb{N}\vDash_{\Sigma_1}(f(x_0,\dots,x_{r\dot{-}1})=x_r)[s,n]).$$

Tenemos que probar que las funciones F_f son recursivas primitivas. Para ello vamos a asociar a cada funtor r-ádico f de $\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner$ un código $c_f\in\mathcal{C}_{\mathrm{rp}}$ que definimos por recursión completa:

1. Si $f\equiv S$, $f\equiv c$ o $f\equiv p^r_i$, definimos, respectivamene, $c_f=\hat{S}$, $c_f=\hat{C}$ o $c_f=\hat{P}^r_i$.
2. Si $f\equiv\kappa(h,g_1,\dots,g_m)$, definimos $c_f=\kappa(c_h,\langle c_{g_1},\dots,c_{g_m}\rangle)$.
3. Si $f\equiv\rho(g,h)$, definimos $c_f=\rho(c_g,c_h)$.

Una simple inducción prueba que, en efecto, $c_f \in \mathcal{C}_{\mathrm{rp}}$ y $\mathrm{Nar}(c_f)$ es el rango del funtor f. Ahora probamos por inducción sobre f que $F_f = \{c_f\}$. Para ello basta probar:

$$f \in \mathrm{Fun}(\ulcorner \mathcal{L}_{\mathrm{arp}} \urcorner) \wedge \mathrm{rang}(f) = r \to$$

$$\bigwedge snc(\ell(s) = r \wedge F_f(s) = n \wedge c = c_f \to \{c\}(s) = n),$$

pues entonces las funciones F_f y $\{c_f\}$ coinciden en $\mathbb{N}^r$. Notemos que la fórmula es de tipo Π_2, pues hemos visto que $F_f(s) = n$ equivale a una fórmula de tipo Π_1, a saber:

$$\bigwedge m(\mathbb{N} \vDash_{\Sigma_1} (f(x_0, \ldots, x_{r\dot{-}1}) = x_r)[s, m] \to m = n).$$

Vamos a detallar únicamente el caso 3 en que $f \equiv \rho(g, h)$, pues el caso 1 es trivial y el caso 2 es mucho más simple. Por hipótesis de inducción tenemos que $F_g = \{c_g\}$ y $F_h = \{c_h\}$. Además, $c_f = \rho(c_g, c_h)$, con lo que $\{c_f\}$ es la función definida por recursión a partir de F_g y F_h. Sólo hay que probar que F_f también lo es. Ahora bien:

$$\underset{\mathrm{ARP}}{\vdash} \bigvee u(f(x_0, \ldots, x_{r\dot{-}1}, 0) = u \wedge g(x_0, \ldots, x_{r\dot{-}1}) = u),$$

luego la traducción de esta fórmula a $\ulcorner \mathcal{L}_a \urcorner$ es un teorema de $\mathrm{I}\Sigma_1$ y 6.15 nos da que, para todo $s \in \mathbb{N}^r$,

$$\mathbb{N} \vDash_{\Sigma_1} \bigvee u(f(x_0, \ldots, x_{r\dot{-}1}, 0) = u \wedge g(x_0, \ldots, x_{r\dot{-}1}) = u)[s],$$

y esto equivale a que $F_f(s, 0) = F_g(s)$. Similarmente,

$$\underset{\mathrm{ARP}}{\vdash} \bigvee uv(f(x_0, \ldots, x_{r\dot{-}1}, Sx_r) = u \wedge f(x_0, \ldots, x_r) = v \wedge h(x_0, \ldots, x_r, v) = u),$$

de donde, fijados $s \in \mathbb{N}^r$ y n, deducimos que existen m, k tales que

$$F_f(s, n+1) = m \wedge F_f(s, n) = k, \wedge F_h(s, n, k) = m,$$

que es lo mismo que $F_f(s, n+1) = F_h(s, n, F_f(s, n))$, luego, en efecto, F_f es la función definida por recursión a partir de F_g y F_h. ∎

Análogamente podemos asociar a cada función $F : \mathbb{N}^r \longrightarrow \mathbb{N}$ recursiva primitiva un funtor f que cumpla el teorema anterior.

8.2 Caracterización aritmética

La definición de las funciones recursivas a partir de códigos es un tanto técnica y laboriosa, pero ello se debe a que contiene la demostración de un teorema no trivial. Vamos a introducir algunos conceptos para enunciarlo con naturalidad.

Definición 8.16 Diremos que una relación $R \subset \mathbb{N}^r$ es de tipo Δ_0, Σ_1 o Π_1 si existe una fórmula $\alpha \in \mathrm{Form}(\ulcorner\mathcal{L}_a\urcorner)$ del tipo correspondiente con r variables (es decir, con variables libres entre $x_0, \ldots, x_{r\dot{-}1}$) tal que

$$\bigwedge s \in \mathbb{N}^r (R(s) \leftrightarrow \mathbb{N} \vDash_0 \alpha[s])$$

(en el caso Δ_0, y con $\vDash_{\Sigma_1}$ o $\vDash_{\Pi_1}$ para los otros dos casos).

Diremos que R es de tipo Δ_1 si existen fórmulas $\alpha, \beta \in \mathrm{Form}(\ulcorner\mathcal{L}_a\urcorner)$ con r variables, de tipo Σ_1 y Π_1, respectivamente, tales que, para todo $s \in \mathbb{N}^r$,

$$R(s) \leftrightarrow \mathbb{N} \vDash_{\Sigma_1} \alpha[s] \leftrightarrow \mathbb{N} \vDash_{\Pi_1} \beta[s].$$

Diremos que una función $F : \mathbb{N}^r \longrightarrow \mathbb{N}$ es de tipo $\Delta_0, \Sigma_1, \Pi_1, \Delta_1$ si lo es su gráfica $G(F)$.

Observemos que si $\phi(s)$ es una fórmula de $\mathcal{L}_a$ de tipo Σ_1 con una variable libre, podemos considerar la fórmula $\alpha = \ulcorner\phi(\langle x_0, \ldots, x_{r\dot{-}1}\rangle)\urcorner \in \mathrm{Form}(\ulcorner\mathcal{L}_a\urcorner)$, de modo que

$$\bigwedge s \in \mathbb{N}^r (\phi(s) \leftrightarrow \mathbb{N} \vDash_{\Sigma_1} \alpha[s]),$$

y análogamente para fórmula de tipo Π_1, por lo que una relación definida por fórmulas de $\mathcal{L}_a$ de tipo Σ_1 o Π_1, o ambas a la vez, es de tipo Σ_1, Π_1 o Δ_1.

Diremos que una relación $R \subset \mathbb{N}^r$ es *recursiva* si lo es su función característica $\chi_R : \mathbb{N}^r \longrightarrow \mathbb{N}$, dada por

$$\chi_R(s) = \begin{cases} 1 & \text{si} \quad R(s), \\ 0 & \text{si} \ \neg R(s). \end{cases}$$

El teorema 8.8 implica ahora trivialmente:

Teorema 8.17 *Toda función recursiva es Δ_1.*

A su vez:

Teorema 8.18 *Toda relación recursiva es Δ_1.*

Demostración: Si $R \subset \mathbb{N}^r$ es una relación recursiva, esto significa que su función característica $\chi_R : \mathbb{N}^r \longrightarrow \{0, 1\}$ es recursiva, luego por el teorema anterior es Σ_1. Esto, a su vez, significa que existe una fórmula $\alpha(x_0, \ldots, x_r)$ de tipo Σ_1 tal que, para todo $s \in \mathbb{N}^r$,

$$R(s) \leftrightarrow \mathbb{N} \vDash_{\Sigma_1} \alpha[s, 1] \leftrightarrow \neg\mathbb{N} \vDash_{\Sigma_1} \alpha[s, 0] \leftrightarrow \mathbb{N} \vDash_{\Pi_1} \neg\alpha[s, 0],$$

luego R es una relación Δ_1. ∎

Vamos a probar el recíproco, para lo cual empezamos demostrando lo siguiente:

Teorema 8.19 (Σ_2-inducción) *Toda relación Δ_0 es recursiva primitiva.*

DEMOSTRACIÓN: En 6.10 definimos el término $\mathrm{Dn}(t,v)$ y en 6.11 la fórmula $\mathbb{N} \vDash_0 \alpha[v]$, ambos de tipo Δ_1 que determinan el número denotado por el semitérmino t de $\ulcorner \mathcal{L}_a \urcorner$ y si la semifórmula α de $\ulcorner \mathcal{L}_a \urcorner$ es satisfecha o no cuando sus variables libres se interpretan según la valoración v, supuesto que v esté definida sobre todas ellas.

Si $s \in \mathbb{N}^r$, $s' \in \mathbb{N}^{r'}$, llamamos

$$v_{s,s'} = \{\langle x_0, s_0\rangle_2, \ldots, \langle x_{r\dot{-}1}, s_{r\dot{-}1}\rangle_2, \langle u_0, s'_0\rangle_2, \ldots, \langle u_{r'\dot{-}1}, s'_{r'\dot{-}1}\rangle_2\}$$

y definimos

$$\mathrm{Dn}(t; s, s') \equiv \mathrm{Dn}(t, v_{s,s'}), \qquad \mathbb{N} \vDash_0 \alpha[s, s'] \equiv \mathbb{N} \vDash_0 \alpha[v_{s,s'}],$$

de modo que si t es un semitérmino y α una semifórmula de $\mathcal{L}_a$ cuyas variables libres estén entre x_i, para $i < r$ y u_i, para $i < r'$, $s \in \mathbb{N}^r$, $s' \in \mathbb{N}^{r'}$, entonces $\mathrm{Dn}(t; s, s')$ es el número denotado por t cuando cada variable x_i se interpreta como s_i y cada variable u_i se interpreta como s'_i, e igualmente $\mathbb{N} \vDash_0 \alpha[s, s']$ significa que la semifórmula α es satisfecha con dichas interpretaciones de sus variables. Es claro que $\mathrm{Dn}(t; s, s')$ sigue siendo un término Δ_1 y $\mathbb{N} \vDash_0 \alpha[s, s']$ sigue siendo una fórmula Δ_1.

Escribiremos $t \in \mathrm{STerm}_{r,r'}(\ulcorner \mathcal{L}_a \urcorner)$ para indicar que t es un semitérmino de $\ulcorner \mathcal{L}_a \urcorner$ cuyas variables (tanto libres como ligadas) estén entre $x_0, \ldots, x_{r\dot{-}1}$, $u_0, \ldots, u_{r'\dot{-}1}$, e igualmente $\alpha \in \mathrm{SForm}^{\Delta_0}_{r,r'}(\ulcorner \mathcal{L}_a \urcorner)$ indicará que α es una semifórmula de $\ulcorner \mathcal{L}_a \urcorner$ de tipo Δ_0 cuyas variables (tanto libres como ligadas) están entre las indicadas.

Definimos como sigue una función recursiva primitiva $D : \mathbb{N}^3 \longrightarrow \mathbb{N}$ por recursión completa sobre t:

1. Si $t \notin \mathrm{STerm}_{r,r'}(\ulcorner \mathcal{L}_a \urcorner)$, entonces $D(r, r', t) = 0$.
2. Si $t \equiv 0$, entonces $D(r, r', t) = \kappa(\hat{P}_0^{r+r'}, \langle \hat{C}\rangle)$.
3. Si $t \equiv x_i$, entonces $D(r, r', t) = \hat{P}_i^{r+r'}$.
4. Si $t \equiv u_i$, entonces $D(r, r', t) = \hat{P}_{r+i}^{r+r'}$.
5. Si $t \equiv St_0$, entonces $D(r, r', t) = \kappa(\hat{S}, \langle D(r, r', t_0)\rangle)$.
6. Si $t \equiv t_1 + t_2$, entonces $D(r, r', t) = \kappa(\hat{+}, \langle D(r, r', t_1), D(r, r', t_2)\rangle)$, donde $\hat{+}$ es el código de la función suma.
7. Si $t \equiv t_1 \cdot t_2$, entonces $D(r, r', t) = \kappa(\hat{\cdot}, \langle D(r, r', t_1), D(r, r', t_2)\rangle)$, donde $\hat{\cdot}$ es el código de la función producto.

La función D es una función concreta determinada por un código $\hat{D} \in \mathcal{C}_{\mathrm{rp}}$ (un numeral) que podemos calcular explícitamente. Una simple inducción sobre t prueba que si $t \in \mathrm{STerm}_{r,r'}(\ulcorner \mathcal{L}_a \urcorner)$, entonces $D(r, r', t) \in \mathcal{C}_{\mathrm{rp}}$ y $\mathrm{Nar}(D(r, r', t)) = r + r'$.

Más aún, de la construcción se sigue fácilmente, también por inducción sobre t, que si $t \in \mathrm{STerm}_{r,r'}(\ulcorner\mathcal{L}_a\urcorner)$, entonces

$$\bigwedge s \in \mathbb{N}^r \bigwedge s' \in \mathbb{N}^{r'} \{D(r,r',t)\}(s^\frown s') = \mathrm{Dn}(t;s,s').$$

Lo único que requiere atención es que la fórmula a la que aplicamos el principio de inducción es de tipo Π_2, pues equivale a

$$t \in \mathrm{Term}_{r,r'}(\ulcorner\mathcal{L}_a\urcorner) \to \bigwedge cnss's''(\{\hat{D}\}(r,r',t) = c \wedge \ell(s) = r \wedge \ell(s') = r' \wedge$$
$$s'' = s^\frown s' \wedge n = \mathrm{Dn}(t;s,s') \to \{c\}(s'') = n).$$

En otros términos, podemos considerar la función $D_t^{r,r'} : \mathbb{N}^{r+r'} \longrightarrow \mathbb{N}$ determinada por que si $s \in \mathbb{N}^r$ y $s' \in \mathbb{N}^{r'}$ entonces $D_t^{r,r'}(s^\frown s') = \mathrm{Dn}(t;s,s')$. Hemos probado que $D_t^{r,r'}$ es recursiva primitiva con código $D(r,r',t)$. Como de costumbre, en ACR_0 podemos probar que cada función $D_t^{r,r'}$, para cada término t en concreto, es recursiva primitiva, y el principio de Σ_2-inducción sólo hace falta para reducir el esquema teoremático a un único teorema.

Ahora definimos una función $\mathrm{St} : \mathbb{N}^3 \longrightarrow \mathbb{N}$ tal que si $\alpha \in \mathrm{SForm}^{\Delta_0}_{r,r'}(\ulcorner\mathcal{L}_a\urcorner)$, entonces $\mathrm{St}(r,r',\alpha)$ sea un código de una función recursiva primitiva con $r+r'$ argumentos, y que dicha función sea la que toma el valor 1 sobre $s^\frown s'$ si se cumple $\mathbb{N} \vDash_0 \alpha[s,s']$ y toma el valor 0 en caso contrario. La definición es como sigue:

1. Si $\alpha \notin \mathrm{SForm}^{\Delta_0}_{r,r'}(\ulcorner\mathcal{L}_a\urcorner)$, definimos $\mathrm{St}(r,r',\alpha) = 0$.

2. Si $\alpha \equiv t_1 = t_2$, consideramos la función

$$F(u,v) = 1 \dot{-} |u - v| = \begin{cases} 1 & \text{si } u = v, \\ 0 & \text{si } u \neq v, \end{cases}$$

   que es recursiva primitiva, y tendrá un código $\hat{F}$, y definimos

$$\mathrm{St}(r,r',\alpha) = \kappa(\hat{F}, \langle\{\hat{D}\}(r,r',t_1), \{\hat{D}\}(r,r',t_2)\rangle).$$

3. Si $\alpha \equiv t_1 \leq t_2$ la definición es análoga, pero considerando la función

$$G(u,v) = 1 \dot{-} (u \dot{-} v) = \begin{cases} 1 & \text{si } u \leq v, \\ 0 & \text{si } u > v. \end{cases}$$

4. Si $\alpha \equiv \neg\beta$, consideramos la función $H(u) = 1 \dot{-} u$ y definimos

$$\mathrm{St}(r,r',\alpha) = \kappa(\hat{H}, \langle\mathrm{St}(r,r',\beta)\rangle).$$

5. Si $\alpha \equiv \beta \vee \gamma$, consideramos

$$I(u,v) = 1 \dot{-} (1 \dot{-} (u+v)) = \begin{cases} 1 & \text{si } u \geq 1 \vee v \geq 1, \\ 0 & \text{si } u = v = 0, \end{cases}$$

   y definimos $\mathrm{St}(r,r',\alpha) = \kappa(\hat{I}, \langle\mathrm{St}(r,r',\beta), \mathrm{St}(r,r',\gamma)\rangle)$.

6. Supongamos ahora que $\alpha \equiv \bigwedge u \leq t\, \beta$, donde $u \equiv u_i$, para cierto $i < r'$ (y la variable u_i no está en t).

   Para cada $s \in \mathbb{N}^{r+r'}$, podemos considerar $s_i[k] \in \mathbb{N}^{r+r'}$ de modo que $s_i[k]_j = s_i$ salvo cuando $j = r + i$, en cuyo caso $s_i[k]_j = k$. En otras palabras, $s_i[k]$ es la misma $r + r'$-tupla s salvo que hemos cambiado por k su componente $r + i$. No podemos decir que la función $s^\frown\langle k\rangle \mapsto s_i[k]$ sea recursiva primitiva porque no hemos definido funciones recursivas primitivas $\mathbb{N}^{r+r'+1} \longrightarrow \mathbb{N}^{r+r'}$, pero sí que son recursivas primitivas sus funciones coordenadas $J^i_j : \mathbb{N}^{r+r'+1} \longrightarrow \mathbb{N}$, que cumplen que $J^i_j(s, k) = s_i$ salvo si $j = r + i$, en cuyo caso $J^i_j(s, k) = k$. Concretamente, $J^i_j = P^{r+r'+1}_i$ salvo si $j = r + i$, en cuyo caso $J^i_j = P^{r+r'+1}_{r+r'}$. Por lo tanto, podemos considerar la $r + r'$-tupla de sus códigos $\langle \hat{J}^i_0, \ldots, \hat{J}^i_{r+r'\dot{-}1}\rangle$.

   Admitiendo que $\mathrm{St}(r, r', \beta)$ es el código de una función recursiva primitiva $\mathrm{Sat} : \mathbb{N}^{r+r'} \longrightarrow \mathbb{N}$, tenemos que $\kappa(\mathrm{St}(r, r', \beta), \langle \hat{J}^i_0, \ldots, \hat{J}^i_{r+r'\dot{-}1}\rangle)$ es el código de la función $s^\frown\langle k\rangle \mapsto \mathrm{Sat}(s_i[k])$, definida en $\mathbb{N}^{r+r'+1}$.

   Ahora consideramos la función $K : \mathbb{N}^{r+r'+1} \longrightarrow \mathbb{N}$ dada por

$$K(s, 0) = 1, \qquad K(s, k + 1) = K(s, k) \cdot \mathrm{Sat}(s_i[k])$$

   o, equivalentemente,

$$K(s, n) = \prod_{k<n} \mathrm{Sat}(s_i[k]).$$

   Por último, consideramos la composición $L(s) = K(s, D^{r,r'}_t(s) + 1)$, es decir,

$$L(s) = \prod_{k \leq D^{r,r'}_t(s)} \mathrm{Sat}(s_i[k]).$$

   Nos ahorramos escribir explícitamente el código de L en términos de $\mathrm{St}(r, r', \beta)$, pero definimos $\mathrm{St}(r, r', \alpha)$ como dicho código.

7. Si $\alpha \equiv \bigvee u \leq t\, \beta$, podríamos hacer una construcción análoga a la anterior, pero es más práctico considerar el código que hemos asociado a $\neg\beta$ a partir de $\mathrm{St}(r, r', \beta)$, luego el código que le hemos asociado a $\bigwedge u \leq t\neg\beta$ en el apartado anterior y, por último, el código que le hemos asignado a $\neg\bigwedge u \leq t\neg\beta$, y tomamos éste como definición de $\mathrm{St}(r, r', \alpha)$ en función de $\mathrm{St}(r, r', \beta)$.

Con esto tenemos definida la función St, que es recursiva primitiva, y le podemos calcular un código explícito (un numeral) $\widehat{\mathrm{St}}$, de modo que $\mathrm{St} = \{\widehat{\mathrm{St}}\}$.

Una simple inducción prueba que si $\alpha \in \mathrm{SForm}^{\Delta_0}_{r,r'}(\ulcorner \mathcal{L}_a \urcorner)$, entonces se cumple que $\mathrm{St}(r, r', \alpha) \in \mathcal{C}_{\mathrm{rp}}$ y que $\mathrm{Nar}(\mathrm{St}(r, r', \alpha)) = r + r'$. Tampoco ofrece dificultad comprobar que si llamamos $\mathrm{Sat}_{r,r'}(\alpha) = \{\mathrm{St}(r, r', \alpha)\} : \mathbb{N}^{r+r'} \longrightarrow \mathbb{N}$, entonces

$$\mathrm{Sat}_{r,r'}(\alpha) : \mathbb{N}^{r+r'} \longrightarrow \{0, 1\}$$

y, si $s \in \mathbb{N}^r$, $s' \in \mathbb{N}^{r'}$, se cumple que

$$\mathrm{Sat}_{r,r'}(\alpha)(s^\frown s') = 1 \leftrightarrow \mathbb{N} \vDash_0 \alpha[s, s'].$$

Se prueba por inducción sobre α, y sólo tenemos que observar que la fórmula a la que aplicamos el principio de inducción es de tipo Π_2:

$$\alpha \in \mathrm{SForm}^{\Delta_0}_{r,r'}(\ulcorner \mathcal{L}_a \urcorner) \to \bigwedge css's''(\{\widehat{\mathrm{St}}\}(r, r', \alpha) = c \wedge \ell(s) = r \wedge \ell(s') = r' \wedge$$

$$s'' = s^\frown s' \wedge (\mathbb{N} \vDash_0 \alpha[s, s'] \to \{c\}(s'') = 1) \wedge (\neg\mathbb{N} \vDash_0 \alpha[s, s'] \to \{c\}(s'') = 0)).$$

Vamos a detallar únicamente el caso en que $\alpha \equiv \bigwedge u \le t\, \beta$. Entonces $u \equiv u_i$ para cierto $i < r'$ y $\beta \in \mathrm{SForm}^{\Delta_0}_{r,r'}(\ulcorner \mathcal{L}_a \urcorner)$. Por hipótesis de inducción tenemos que $\mathrm{Sat}_{r,r'}(\beta) : \mathbb{N}^{r+r'} \longrightarrow \{0, 1\}$ determina si se cumple o no $\mathbb{N} \vDash_0 \beta[s, s']$.

En la primera parte de la prueba hemos visto que $D_t^{r,r'}(s^\frown s') = \mathrm{Dn}(t; s, s')$ y, por construcción,

$$\mathrm{Sat}_{r,r'}(\alpha)(s^\frown s') = \prod_{k \le \mathrm{Dn}(t;s,s')} \mathrm{Sat}_{r,r'}(\beta)((s^\frown s')_i[k]).$$

Como los factores sólo pueden tomar los valores $0, 1$, lo mismo vale para el producto, luego $\mathrm{Sat}_{r,r'}(\alpha) : \mathbb{N}^{r+r'} \longrightarrow \{0, 1\}$. Además, se cumplirá

$$\mathrm{Sat}_{r,r'}(\alpha)(s^\frown s') = 1$$

si y sólo si para todo $k \le \mathrm{Dn}(t; s, s')$ se cumple $\mathrm{Sat}_{r,r'}(\beta)((s^\frown s')_i[k]) = 1$, si y sólo si para todo $k \le \mathrm{Dn}(t; s, s')$, se cumple $\mathbb{N} \vDash_0 \beta[s, s'']$, donde s'' se diferencia de s' en que $s''_i = k$, y es claro que esto equivale a $\mathbb{N} \vDash_0 \bigwedge u \le t\beta[s, s']$.

Ahora observamos que si $\alpha \in \mathrm{SForm}^{\Delta_0}_{r,r'}(\ulcorner \mathcal{L}_a \urcorner)$ es una fórmula, es decir, si no tiene variables libres de tipo ligado, la relación $\mathbb{N} \vDash_0 \alpha[s, s']$ no depende de s', es decir, equivale a la fórmula $\mathbb{N} \vDash_0 \alpha[s]$. Por ello, si definimos

$$\mathrm{Sat}^*_{r,r'}(\alpha) : \mathbb{N}^r \longrightarrow \{0, 1\}$$

como la composición de $\mathrm{Sat}_{r,r'}(\alpha) : \mathbb{N}^{r+r'} \longrightarrow \{0, 1\}$ con las proyecciones

$$\langle P^r_0, \dots, P^r_{r \dot- 1}, P^r_0, \dots, P^r_0 \rangle : \mathbb{N}^r \longrightarrow \mathbb{N}^{r+r'}$$

obtenemos una función recursiva primitiva tal que

$$\mathrm{Sat}^*_{r,r'}(\alpha)(s) = 1 \leftrightarrow \mathbb{N} \vDash_0 \alpha[s].$$

Esto significa que $\mathrm{Sat}^*_{r,r'}(\alpha)$ es la función característica de la relación $R \subset \mathbb{N}^r$ definida por la fórmula α, luego R es una relación recursiva primitiva. ∎

Insistimos en que, si tenemos una fórmula concreta α de tipo Δ_0 con variables libres entre $x_0, \dots, x_{r \dot- 1}$, en ACR_0 podemos probar que la relación $R \subset \mathbb{N}^r$ que define es recursiva sin necesidad del principio de Σ_2-inducción.

A su vez:

Teorema 8.20 ($\boldsymbol{\Sigma_2}$-inducción) *Una relación es recursiva si y sólo si es Δ_1.*

DEMOSTRACIÓN: El teorema 8.18 nos da una implicación. Recíprocamente, si $R \subset \mathbb{N}^r$ es una relación Δ_1, esto significa que existen fórmulas

$$\alpha(x_0, \ldots, x_r), \quad \beta(x_0, \ldots, x_r) \in \mathrm{Form}(\ulcorner \mathcal{L}_a \urcorner)$$

de tipo Δ_0 con $r+1$ variables libres tales que, para todo $s \in \mathbb{N}^r$, se cumple

$$R(s) \leftrightarrow \mathbb{N} \vDash_{\Sigma_1} \bigwedge u \alpha(x_0, \ldots, x_{r \dot{-} 1}, u)[s] \leftrightarrow \mathbb{N} \vDash_{\Pi_1} \bigwedge u \beta(x_0, \ldots, x_{r \dot{-} 1}, u)[s].$$

Por el teorema anterior, las fórmulas α y β definen relaciones recursivas en $\mathbb{N}^{r+1}$. Llamamos $F, G : \mathbb{N}^{r+1} \longrightarrow \mathbb{N}$ a sus funciones características, que son funciones recursivas primitivas. Así,

$$R(s) \leftrightarrow \bigwedge u\, F(s,u) = 1 \leftrightarrow \bigvee u\, G(s,u) = 1.$$

Consideramos la función recursiva primitiva $H : \mathbb{N}^{r+1} \longrightarrow \mathbb{N}$ dada por

$$H(s,u) = F(s,u) \cdot (1 \dot{-} G(s,u)).$$

Vemos así que si se cumple $R(s)$, existe un u que cumple $G(s,u) = 1$, y eso hace que $H(s,u) = 0$, mientras que si $\neg R(s)$, existe un u tal que $F(s,u) = 0$, y también $H(s,u) = 0$, luego

$$\bigwedge s \in \mathbb{N}^r \bigvee u\, H(s,u) = 0,$$

y así podemos definir por minimización la función recursiva $I : \mathbb{N}^r \longrightarrow \mathbb{N}$ dada por

$$I(s) = \mu u\, H(s,u) = 0.$$

Claramente, si $I(s) = u$, se cumple que $F(s,u) = 0$ (en cuyo caso $\neg R(s)$) o bien $G(s,u) = 1$ (en cuyo caso $R(s)$), luego

$$R(s) \leftrightarrow G(s, I(s)) = 1.$$

La función $J(s) = G(s, I(s))$ es recursiva, y claramente es la función característica χ_R, luego la relación R es recursiva. ∎

Teorema 8.21 ($\boldsymbol{\Sigma_2}$-inducción) *Una función es recursiva si y sólo si es Σ_1, si y sólo si es Δ_1.*

DEMOSTRACIÓN: Por el teorema 8.17, si una función es recursiva, entonces es Δ_1. Recíprocamente, supongamos que una función $F : \mathbb{N}^r \longrightarrow \mathbb{N}$ es Δ_1. Esto significa que su gráfica $G(F)$ es una relación Δ_1, luego recursiva. Sea $J : \mathbb{N}^{r+1} \longrightarrow \{0,1\}$ su función característica, de modo que

$$\bigwedge s \in \mathbb{N}^r \bigwedge n (F(s) = n \leftrightarrow J(s,n) = 1).$$

En particular $\bigwedge s \in \mathbb{N}^r \bigvee n\, 1 \dot{-} J(s,n) = 0$, y

$$F(s) = \mu n(1 \dot{-} J(s,n) = 0),$$

por lo que F es una función recursiva.

Obviamente, toda función Δ_1 es Σ_1 y, recíprocamente, si $F : \mathbb{N}^r \longrightarrow \mathbb{N}$ es Σ_1, esto significa que existe una fórmula α de $\mathcal{L}_a$ con $r+1$ variables libres tal que, para todo $t \in \mathbb{N}^{r+1}$,

$$t \in G(F) \leftrightarrow \mathbb{N} \vDash_{\Sigma_1} \alpha[t].$$

Entonces, teniendo en cuenta que, para todo $s \in \mathbb{N}^r$, se cumple

$$F(s) = n \leftrightarrow \bigwedge m(F(s) = m \to m = n),$$

vemos que, para todo $t \in \mathbb{N}^{r+1}$,

$$t \in G(F) \leftrightarrow \bigwedge m(\mathbb{N} \vDash_{\Sigma_1} \alpha[t|_r, m] \to m = t_r).$$

La fórmula de la derecha es Π_1, y esto implica que $G(F)$ (luego F) es Π_1. ∎

8.3 Funciones recursivas parciales

Hemos indicado al principio de este capítulo que las funciones recursivas son las funciones que puede calcular un ordenador que ejecute un algoritmo prefijado. En esta sección vamos a analizar esto con más detalle y, paradójicamente, veremos que para estudiar las funciones, relaciones y conjuntos *computables,* es decir, que pueden determinarse mediante algoritmos, es indispensable hablar de funciones, relaciones y conjuntos no computables, lo que nos obliga a ir más allá de ACR_0 (que no permite definir más que objetos estrictamente finitistas y, por consiguiente, computables) para trabajar en ACA_0.

Es frecuente que un algoritmo requiera ejecutar varias veces un mismo bloque de instrucciones. Esto es lo que se denomina un *bucle,* pero hay dos tipos de bucles sustancialmente distintos. Unos son los que, en los distintos lenguajes de programación, se expresan con instrucciones de tipo FOR $i = 1, \ldots, n$, de modo que el número de veces que el bucle tiene que repetirse es conocido desde el principio, pero también hay bucles de tipo WHILE, que se tienen que ejecutar hasta que se cumpla una condición. Cualquier programador sabe que los bucles de tipo WHILE encierran un peligro, y es la posibilidad de que la condición que ponga fin al bucle nunca llegue a cumplirse, con lo que el algoritmo nunca termina y el ordenador "se queda colgado", y esto puede depender de los argumentos de los que parte el algoritmo, de modo que puede suceder que un algoritmo pueda calcular el valor de una función para ciertos argumentos dados y que "se quede colgado" cuando los argumentos son otros.

Esto se traduce en que si queremos estudiar las funciones que pueden calcularse mediante un algoritmo, tenemos que contemplar la posibilidad de que no estén definidas para algunos valores de los argumentos de la función, para los cuales el algoritmo no termina nunca. Esto nos lleva al concepto siguiente:

Definición 8.22 Una función $F : \mathbb{N}^r \longrightarrow \mathbb{N}$ *parcial* es un conjunto $F \subset \mathbb{N}^r \times \mathbb{N}$ tal que

$$\bigwedge smn(\langle s, m\rangle_2 \in F \wedge \langle s, n\rangle_2 \in F \to m = n).$$

Equivalentemente, considerando el dominio $\mathcal{D}F = \{s \mid \bigvee n\ \langle s, n\rangle_2 \in F\}$, vemos que las funciones r-ádicas parciales son los conjuntos F cuyo dominio cumple $\mathcal{D}F \subset \mathbb{N}^r$ y además $F : \mathcal{D}F \longrightarrow \mathbb{N}$.

En particular, toda función $F : \mathbb{N}^r \longrightarrow \mathbb{N}$ es una función r-adica parcial, con la peculiaridad de que $\mathcal{D}F = \mathbb{N}^r$, y en tal caso diremos que se trata de una función $F : \mathbb{N}^r \longrightarrow \mathbb{N}$ *total.*

Para mantener el convenio usual, cuando hablemos de funciones, se entenderá que son funciones totales, de manera que las funciones $F : \mathbb{N}^r \longrightarrow \mathbb{N}$ (totales) son un caso particular de las funciones parciales $F : \mathbb{N}^r \longrightarrow \mathbb{N}$.

Los tres procedimientos de construcción de funciones recursivas (composición, recursión y minimización) pueden extenderse para tratar con funciones parciales:

Teorema 8.23 (Composición parcial) *Dado un conjunto G para el que se cumpla $\bigwedge i < m\, G_i : \mathbb{N}^n \longrightarrow \mathbb{N}$ parcial y una función parcial $H : \mathbb{N}^m \longrightarrow \mathbb{N}$, existe una única función parcial $F : \mathbb{N}^n \longrightarrow \mathbb{N}$ tal que F está definida en $s \in \mathbb{N}^n$ si y sólo si $G_0, \ldots, G_{m-1}$ están definidas en s y H está definida en $\langle G_0(s), \ldots, G_{m-1}(s)\rangle$, y en tal caso*

$$F(s) = H(G_0(s), \ldots, G_{m-1}(s)).$$

En estas condiciones diremos que F está definida por *composición parcial* a partir de G y H.

Demostración: Basta definir, por comprensión aritmética,

$$F = \{z \mid \bigvee str(z = \langle s, r\rangle_2 \wedge \ell(s) = n \wedge \ell(t) = m \wedge \bigwedge i < m\, \langle s, t_i\rangle_2 \in G_i) \wedge \langle t, r\rangle_2 \in H\}.$$

∎

Teorema 8.24 (Recursión parcial) *Si $G : \mathbb{N}^m \longrightarrow \mathbb{N}$ y $H : \mathbb{N}^{m+2} \longrightarrow \mathbb{N}$, con $m > 0$ son funciones parciales, existe una única función $F : \mathbb{N}^{m+1} \longrightarrow \mathbb{N}$ parcial tal que, para todo $s \in \mathbb{N}^{m+1}$ se cumple que F está definida en s si y sólo si:*

1. *G está definida en $s|_m$, F está definida en $s|_m{}^\frown\langle 0\rangle$ y*
$$F(s|_m{}^\frown\langle 0\rangle) = G(s|_n).$$
2. *Para cada $i < s_m$, F está definida en $s|_m{}^\frown\langle i\rangle$ y en $s|_m{}^\frown\langle i+1\rangle$, G está definida en $s|_m$ y H está definida en $s|_m{}^\frown\langle i, F(s|_m{}^\frown\langle i\rangle)\rangle$ y*
$$F(s|_m, i+1) = H(s|_m, i, F(s|_m, i)).$$

En el caso $m = 0$ sustituimos la función G por un número a y cambiamos la condición 1 por que F esté definida en 0 y que $F(0) = a$.

En estas condiciones diremos que F está definida por *recursión parcial* a partir de G y H (o de a y H).

DEMOSTRACIÓN: Basta definir

$$F = \{z \mid \bigvee st(\ell(s) = m+1 \wedge \ell(t) = s_m + 1 \wedge z = \langle s, t_{s_m}\rangle_2 \wedge$$

$$\langle s|_m, t_0\rangle_2 \in G \wedge \bigwedge i < s_m\ \langle s|_m{}^\frown\langle i, t_i\rangle, t_{i+1}\rangle_2 \in H)\}.$$

El caso $m = 0$ se trata de forma similar. ∎

Teorema 8.25 (Minimización parcial) *Si $G : \mathbb{N}^{n+1} \longrightarrow \mathbb{N}$ es una función parcial, existe una única función parcial $F : \mathbb{N}^n \longrightarrow \mathbb{N}$ tal que, para todo $s \in \mathbb{N}^n$, se cumple que F está definida en s si y sólo si existe un número natural r tal que:*

1. *Para todo $i \leq r$ se cumple que G está definida en $s^\frown\langle i\rangle$.*
2. *Para todo $i < r$ se cumple que $G(s, i) \neq 0$.*
3. *$G(s, r) = 0$.*

y en tal caso $F(s) = r$.

En estas condiciones diremos que F está definida por *minimización parcial* a partir de G.

DEMOSTRACIÓN: Basta definir

$$F = \{z \mid \bigvee sr(\ell(s) = n \wedge z = \langle s, r\rangle \wedge$$

$$\bigwedge i < r \bigvee k(k \neq 0 \wedge \langle s^\frown\langle i\rangle, k\rangle_2 \in G) \wedge \langle s^\frown\langle r\rangle, 0\rangle_2 \in G)\}.$$ ∎

Así, al definir F a partir de G por minimización parcial no exigimos que exista un mínimo r tal que $G(s, r) \neq 0$, sino que, en caso de que no exista (o en caso de que al ir calculando $G(s, 0), G(s, 1), \ldots$ lleguemos a un i tal que $G(s, i)$ no esté definido antes de encontrar un r tal que $G(s, r) = 0$), la función F no está definida en s, pero eso no pone en cuestión que F esté definida por minimización parcial a partir de G.

En la definición 8.5 hemos definido los *códigos de funciones recursivas parciales*. Posteriormente nos hemos referido a ellos simplemente como códigos, pero ahora podemos dar sentido a su nombre completo:

Definición 8.26 Si $c \in \mathcal{C}$ es un código de una función recursiva parcial, llamamos

$$\{c\} = \{z \mid \bigvee sn(z = \langle s, n\rangle_2 \wedge \ell(s) = \mathrm{Nar}(c) \wedge \{c\}(s) = n)\}.$$

Notemos que el conjunto $\{c\}$ es Σ_1, no necesariamente Δ_1, por lo que necesitamos el axioma de comprensión aritmética para garantizar su existencia.

El teorema 8.6 nos da que, si $\mathrm{Nar}(c) = r$, entonces $\{c\} : \mathbb{N}^r \longrightarrow \mathbb{N}$ es una función parcial. A las funciones de esta forma las llamaremos *funciones recursivas parciales.*

Notemos que, por definición, las funciones recursivas son las funciones recursivas parciales que son, de hecho, totales.

Ahora todo código $c \in \mathcal{C}$ define una función recursiva, pero ésta puede ser parcial. Una ligera variante de la demostración de 8.4 nos da la versión siguiente del teorema 8.9:

Teorema 8.27 *Se cumple:*

1. *Los códigos $\hat{S}, \hat{C}, \hat{P}^n_j$ cumplen $\{\hat{S}\} = S$, $\{\hat{C}\} = C$, $\{\hat{P}^n_j\} = P^n_j$.*

2. *Si $c_1, \ldots, c_m$ son códigos de funciones n-ádicas recursivas parciales, c' es un código de una función m-ádica recursiva parcial y $c = \kappa(c', \langle c_1, \ldots, c_m\rangle)$, entonces $\{c\} : \mathbb{N}^n \longrightarrow \mathbb{N}$ es la función parcial definida por composición parcial a partir de $\{c'\} : \mathbb{N}^m \longrightarrow \mathbb{N}$ y $\{c_i\} : \mathbb{N}^n \longrightarrow \mathbb{N}$.*

3a. *Si c' es un código de una función n-ádica recursiva parcial, c'' es un código de una función $n+2$-ádica recursiva parcial y $c = \rho(c', c'')$, entonces $\{c\} : \mathbb{N}^{n+1} \longrightarrow \mathbb{N}$ es la función parcial definida por recursión parcial a partir de $\{c'\} : \mathbb{N}^n \longrightarrow \mathbb{N}$ y $\{c''\} : \mathbb{N}^{n+2} \longrightarrow \mathbb{N}$.*

3b *Si c' es un código de una función diádica recursiva parcial y $c = \rho^*(a, c')$, entonces $\{c\} : \mathbb{N} \longrightarrow \mathbb{N}$ es la función parcial definida por recursión parcial a partir de a y de $\{c'\} : \mathbb{N}^2 \longrightarrow \mathbb{N}$.*

4. *Si c' es el código de una función $n+1$-ádica recursiva parcial y $c = \mu(c')$, entonces $\{c\} : \mathbb{N}^n \longrightarrow \mathbb{N}$ es la función parcial definida por minimización parcial a partir de $\{c'\} : \mathbb{N}^{n+1} \longrightarrow \mathbb{N}$.*

Este teorema muestra que las funciones recursivas parciales son las que pueden definirse a partir de las funciones recursivas elementales mediante composición parcial, recursión parcial o minimización parcial, pero en ACA_0 podemos formular esto explícitamente:

Teorema 8.28 *Una función parcial $F : \mathbb{N}^r \longrightarrow \mathbb{N}$ es recursiva parcial si y sólo si existen l, k tales que $\ell(k) = l + 1$, $\bigwedge i \le l\, k_i \ge 1$ y un conjunto H tal que $\bigwedge i \le l\ H_i : \mathbb{N}^{k_i} \longrightarrow \mathbb{N}$ parcial, $H_l = F$ y, para cada $i \le l$, se cumple uno de los casos siguientes:*

1. *$H_i = S$, $H_i = C$ o $\bigvee j < k_i\, H_i = P^{k_i}_j$.*

2. *Existen $j < i$ y s tales que $\ell(s) = k_j$, $\bigwedge u < k_j\, (s_u < i \wedge k_{s_u} = k_i)$ de modo que H_i está definida por composición parcial a partir de la sucesión $\langle H_{s_0}, \ldots, H_{s_{k_j \dot{-} 1}}\rangle$ y de H_j.*

3a. Existen $j, l < i$ tales que $k_i = k_j + 1$, $k_l = k_j + 2$, y H_i es la función definida por recursión parcial a partir de H_j y H_l.

3b. Existen $j < i$ y a tales que $k_i = 1$, $k_j = 2$ y H_i es la función definida por recursión parcial a partir de a y H_j.

4. Existe un $j < i$ tal que $k_j = k_i + 1$ y H_i es la función definida por minimización parcial a partir de H_j.

DEMOSTRACIÓN: Si existe H, una simple inducción sobre i prueba que cada H_i es recursiva parcial, luego en particular lo es $F = H_l$. (Notemos que la fórmula a la que le aplicamos el principio de inducción es aritmética, pues sólo contiene la variable de segundo orden H.)

Para probar el recíproco observamos en primer lugar que si $c \in \mathcal{C}$ es un código para una función recursiva parcial existe un $\bar{c}$ tal que $\ell(\bar{c}) = l+1$, $\bar{c}_l = c$ y, para cada $i < l$, se cumple uno de los casos siguientes:

1. $\bar{c}_i = \hat{S}$, $\bar{c}_i = \hat{C}$ o existen $j < n$ tales que $\bar{c}_i = \hat{P}^n_j$.
2. Existen $j < i$ y s tales que $\ell(s) = \mathrm{Nar}(\bar{c}_j) = k$,
$$\bigwedge u < \ell(s)(s_u < i \wedge \mathrm{Nar}(\bar{c}_{s_u}) = \mathrm{Nar}(\bar{c}_i)),$$
y $\bar{c}_i = \kappa(\bar{c}_j, \langle \bar{c}_{s_0}, \ldots, \bar{c}_{s_k-1}\rangle)$.

3a. Existen $j, l < i$ tales que $\mathrm{Nar}(\bar{c}_i) = \mathrm{Nar}(\bar{c}_j) + 1$, $\mathrm{Nar}(\bar{c}_l) = \mathrm{Nar}(\bar{c}_j) + 2$ y $\bar{c}_i = \rho(\bar{c}_j, \bar{c}_l)$.

3b. Existen $j < i$ y a tales que $\mathrm{Nar}(\bar{c}_i) = 1$, $\mathrm{Nar}(\bar{c}_j) = 2$ y $\bar{c}_i = \rho^*(a, \bar{c}_j)$.

4. Existe un $j < i$ tal que $\mathrm{Nar}(\bar{c}_j) = \mathrm{Nar}(\bar{c}_i) + 1$ y $\bar{c}_i = \mu(\bar{c}_j)$.

Esto se prueba trivialmente por inducción sobre c a partir de la definición de código.

Como consecuencia, si F es recursiva parcial, es de la forma $F = \{c\}$, para cierto $c \in \mathcal{C}$, tomamos $\bar{c}$ que cumpla lo anterior y definimos

$$H = \{z \mid \bigvee i \le l \bigvee sn(z = \langle i, \langle s, n\rangle_2\rangle_2 \wedge \ell(s) = \mathrm{Nar}(\bar{c}_i) \wedge \{\bar{c}_i\}(s) = n)\}.$$

De este modo $H_i = \{\bar{c}_i\}$ y se cumple lo requerido. ∎

Similarmente:

Teorema 8.29 *Una función parcial $F : \mathbb{N}^r \longrightarrow \mathbb{N}$ es recursiva (primitiva) si y sólo si existen l, k tales que $\ell(k) = l + 1$, $\bigwedge i \le l\, k_i \ge 1$ y un conjunto H tal que $\bigwedge i \le l\ H_i : \mathbb{N}^{k_i} \longrightarrow \mathbb{N}$, $H_l = F$ y, para cada $i \le l$, se cumple uno de los casos siguientes:*

1. $H_i = S$, $H_i = C$ o $\bigvee j < k_i\, H_i = P^{k_i}_j$.

2. *Existen $j < i$ y s tales que $\ell(s) = k_j$, $\bigwedge u < k_j\,(s_u < i \wedge k_{s_u} = k_i)$ de modo que H_i está definida por composición a partir de la sucesión $\langle H_{s_0}, \ldots, H_{s_{k_j \dot{-} 1}}\rangle$ y de H_j.*

3a. *Existen $j, l < i$ tales que $k_i = k_j + 1$, $k_l = k_j + 2$, y H_i es la función definida por recursión a partir de H_j y H_l.*

3b. *Existen $j < i$ y a tales que $k_i = 1$, $k_j = 2$ y H_i es la función definida por recursión a partir de a y H_j.*

4. *(Sólo para el caso de funciones recursivas) Existe un $j < i$ de manera que $k_j = k_i + 1$, $\bigwedge s \in \mathbb{N}^{k_j} \bigvee v\, H(s, v) = 0$ y H_i es la función definida por minimización a partir de H_j.*

DEMOSTRACIÓN: El caso correspondiente a las funciones recursivas primitivas se prueba igual que el teorema anterior, sin más que tener en cuenta que los códigos de funciones recursivas primitivas definen siempre funciones totales (teorema 8.10).

En cambio, el caso de las funciones recursivas es más sutil, porque, en principio, si $F = \{c\}$ es una función recursiva, sólo tenemos garantizado que el código c define una función total, pero no que los códigos a partir de los que se construye c definan funciones totales. Ahora, bien, si F es una función recursiva, por 8.18 sabemos que es Δ_1, luego $G(F)$ es una relación Δ_1, y en la prueba del teorema 8.20 hemos visto que existen funciones recursivas primitivas G, H a partir de las cuales se define otra función I por minimización y luego J por composición, de modo que

$$t \in G(F) \leftrightarrow J(t) = 1.$$

Por último, en la prueba de 8.21 se ve que F se define por minimización a partir de $1 \dot{-} J$.

Por consiguiente, una sucesión de funciones en las condiciones del enunciado para las funciones recursivas primitivas G, H y $1 \dot{-} s$ (que ya sabemos que existe), se completa con cuatro pasos más para obtener F. ∎

Las funciones recursivas parciales tienen una caracterización aritmética análoga al teorema 8.21:

Teorema 8.30 *Una función parcial es recursiva parcial si y sólo si es Σ_1.*

DEMOSTRACIÓN: Si una función parcial $F : \mathbb{N}^r \longrightarrow \mathbb{N}$ es recursiva parcial, existe un código $c \in \mathcal{C}$ tal que $F = \{c\}$, luego, por la propia definición 8.26, tenemos que $G(F)$ es Σ_1.

Supongamos ahora que F es Σ_1. Esto significa que existe una fórmula $\alpha(x_0, \ldots, x_r, x_{r+1}) \in \mathrm{Form}(\ulcorner \mathcal{L}_a \urcorner)$ de tipo Δ_0 tal que

$$\bigwedge s \in \mathbb{N}^{r+1}(s \in G(F) \leftrightarrow \mathbb{N} \vDash_{\Sigma_1} \bigvee u\, \alpha(x_0, \ldots, x_r, u)[s]).$$

Por 8.20 tenemos que la relación $R \subset \mathbb{N}^{r+2}$ dada por

$$\bigwedge t \in \mathbb{N}^{r+2}(R(t) \leftrightarrow \mathbb{N} \vDash_{\Sigma_1} \alpha[t])$$

es recursiva, lo que a su vez significa que $\chi_R : \mathbb{N}^{r+2} \longrightarrow \mathbb{N}$ es recursiva. En estos términos,

$$\bigwedge s \in \mathbb{N}^{r+1}(s \in G(F) \leftrightarrow \bigvee u\, \chi_R(s,u) = 1).$$

Definimos la función recursiva parcial $H(s) = \mu u\,(1 \dot{-} \chi_R(s,u)) = 0$, de modo que $H(s)$ está definida si y sólo si $s \in G(F)$. Si componemos H con la función constante 0, obtenemos otra función recursiva parcial $H'(s) = c_0(H(s))$ que vale 0 cuando está definida, pero si $H(s)$ no está definida, entonces H' tampoco. Finalmente, observamos que, para cada $x \in \mathbb{N}^r$,

$$F(x) = \mu n\, H'(x,n) = 0,$$

pues el miembro derecho (si está definido) es el mínimo número natural tal que la función $H(x,n)$ está definida, y en tal caso es el único número natural tal que $x^\frown\langle n\rangle \in G(F)$, luego es $F(x)$. Esto prueba que F es recursiva parcial. ∎

La tesis de Church-Turing La caracterización de las funciones recursivas parciales dada por el teorema 8.28 (con la posibilidad de omitir el punto 3b) es la forma más habitual de definir las funciones recursivas parciales, y es especialmente idónea para demostrar la llamada *tesis de Church-Turing,* según la cual las funciones recursivas parciales son exactamente las funciones que pueden calcularse mediante un algoritmo, contemplando la posibilidad de que éste no termine y deje algunos valores indefinidos.

Para demostrar la tesis de Church-Turing se usa habitualmente el concepto de *máquina de Turing,* que es un modelo de ordenador ideal con una memoria potencialmente infinita. Por una parte se demuestra que toda función recursiva parcial puede calcularse mediante una máquina de Turing y, por otro, que toda función calculable mediante una máquina de Turing es recursiva parcial, pero de modo que el argumento es claramente adaptable a cualquier otro "ordenador" en el sentido más amplio del término, es decir, a cualquier dispositivo que pueda leer y escribir datos en una memoria potencialmente infinita y que los pueda manipular mediante un proceso determinista. Para los detalles remitimos a las secciones [LM 7.4, 7.5], puesto que el argumento tiene poco que ver con los resultados que estamos presentando aquí. Recalcamos únicamente que la incorporación de la definición por minimización a la definición de las funciones recursivas parciales tiene el efecto de reflejar la posibilidad de que un algoritmo entre en un bucle, que en este caso sería el que va calculando

$$f(x_1,\ldots,x_n,0), \quad f(x_1,\ldots,x_n,1), \quad f(x_1,\ldots,x_n,2), \quad \ldots$$

con un criterio de finalización que *a priori* no tiene por qué darse nunca (en este caso, que una de las funciones tome el valor 0 antes de que llegar a alguna que no esté definida), y que, por consiguiente, deja abierta la posibilidad de que el algoritmo no termine, y a la vez de que no podamos saber si todavía no ha terminado, pero terminará más adelante. ∎

Funciones universales Una función recursiva parcial notable es la función parcial $\mathcal{U} : \mathbb{N}^2 \longrightarrow \mathbb{N}$ que está definida sobre los pares $\langle c, s\rangle$ tales que $c \in \mathcal{C}$, $\mathrm{Nar}(c) = r$, $s \in \mathbb{N}^r$ y $\{c\}(s)\downarrow$, y en tal caso $\mathcal{U}(c, s) = \{c\}(s)$. Explícitamente:

$$\mathcal{U} = \{z \mid \bigvee csn(z = \langle\langle c, s\rangle, n\rangle_2 \wedge c \in \mathcal{C} \wedge \ell(s) = \mathrm{Nar}(c) \wedge \{c\}(s) = n)\}.$$

Claramente $\mathcal{U}$ es una función parcial de tipo Σ_1, luego es recursiva parcial, y es lo que se conoce como una *función recursiva parcial universal,* en el sentido de que permite calcular cualquier función recursiva parcial: Si $\{c\}$ es una función recursiva parcial, entonces $\{c\}(s)$ está definido si y sólo si lo está $\mathcal{U}(c, s)$, y en tal caso ambas funciones toman el mismo valor.

Similarmente podemos definir una función $\mathcal{U}_{\mathrm{rp}} : \mathbb{N}^2 \longrightarrow \mathbb{N}$ mediante

$$\mathcal{U}_{\mathrm{rp}} = \{z \mid \bigvee csn(z = \langle\langle c, s\rangle, n\rangle_2 \wedge ((c \in \mathcal{C}_{\mathrm{rp}} \wedge \ell(s) = \mathrm{Nar}(c) \wedge \{c\}(s) = n) \vee$$
$$((c \notin \mathcal{C}_{\mathrm{rp}} \vee \ell(s) \neq \mathrm{Nar}(c)) \wedge n = 0)))\}.$$

Teniendo en cuenta que si $c \in \mathcal{C}_{\mathrm{rp}} \wedge \ell(s) = \mathrm{Nar}(c)$ no puede darse el caso de que $\{c\}(s)\uparrow$, tenemos que $\mathcal{U}_{\mathrm{rp}}$ es una función total, y también es Σ_1, luego es recursiva.

Podemos pensar que $\mathcal{U}$ y $\mathcal{U}_{\mathrm{rp}}$ son funciones "programables", en el sentido de que les damos un "programa" (un código de una función recursiva parcial o recursiva primitiva) y unos argumentos adecuados a dicho programa, y nos ejecuta el programa sobre dichos argumentos.

De aquí podemos obtener algunos ejemplos interesantes:

Un ejemplo de función recursiva no recursiva primitiva Sea $\mathcal{C}^1_{\mathrm{rp}}$ el conjunto de los códigos de las funciones monádicas recursivas primitivas:

$$\mathcal{C}^1_{\mathrm{rp}} = \{c \mid c \in \mathcal{C}_{\mathrm{rp}} \wedge \mathrm{Nar}(c) = 1\}.$$

Se trata de un conjunto infinito, luego, por el teorema 7.23 existe una biyección creciente $\pi : \mathbb{N}^1 \longrightarrow \mathcal{C}^1_{\mathrm{rp}}$. Esta biyección es recursiva primitiva. En efecto, la relación dada por

$$R(x, y) \leftrightarrow x < y \wedge y \in \mathcal{C}^1_{\mathrm{rp}}$$

es Δ_0, luego es recursiva primitiva, luego su función característica $\chi_R : \mathbb{N}^2 \longrightarrow \mathbb{N}$ también lo es. Si $c \in \mathcal{C}^1_{\mathrm{rp}}$, es claro que lo mismo le sucede a $c' = \kappa(\hat{C}, c)$ y $c < c'$, luego la función

$$M(c) = \mu u \leq \kappa(\hat{C}, c)\,(1 \dot{-} \chi_R(c, u)) = 0$$

es recursiva primitiva, y si $c \in \mathcal{C}^1_{\mathrm{rp}}$, entonces $M(c)$ es el menor código de $\mathcal{C}^1_{\mathrm{rp}}$ mayor que c. A su vez, esto nos permite definir

$$\pi^1(0) = \hat{S}, \qquad \pi^1(m + 1) = M(\pi^1(m)),$$

que es claramente la biyección creciente $\pi : \mathbb{N}^1 \longrightarrow \mathcal{C}^1_{\mathrm{rp}}$.

Ahora consideramos la función $F : \mathbb{N}^2 \longrightarrow \mathbb{N}$ dada por

$$F(m,n) = \{\pi^1(m)\}(n) = \mathcal{U}_{\mathrm{rp}}(\pi^1(m), n).$$

En otros términos las funciones $\{\pi^1(m)\}$ recorren (con repeticiones) todas las funciones monádicas recursivas primitivas y $F(m,n)$ es el valor que toma en n la función monádica recursiva primitiva m-sima.

Tenemos que F es recursiva por ser composición de una función recursiva y otra recursiva primitiva. Ahora bien, de aquí deducimos que $\mathcal{U}_{\mathrm{rp}}$ no puede ser recursiva primitiva, pues de serlo también lo sería F, y a su vez lo sería la función $G(n) = F(n,n) + 1$, que tendría un código $c = \pi^1(m)$, para cierto m, con lo que $\bigwedge n\, G(n) = F(m,n)$, y en particular $F(n,n) + 1 = G(n) = F(n,n)$.

Así pues, la función recursiva primitiva universal es recursiva, pero no recursiva primitiva. ∎

Más interesante es el ejemplo siguiente:

Un ejemplo de función no recursiva Ahora consideramos el conjunto $\mathcal{C}_r^1$ formado por los códigos de funciones monádicas recursivas, es decir,

$$\mathcal{C}_r^1 = \{c \mid c \in \mathcal{C} \wedge \mathrm{Nar}(c) = 1 \wedge \bigwedge n \bigvee m\, \{c\}(\langle n \rangle) = m\}.$$

Nuevamente, como $\mathcal{C}_r^1$ es infinito, el teorema 7.23 nos da una biyección creciente $\pi : \mathbb{N}^1 \longrightarrow \mathcal{C}_r^1$, que a su vez nos permite definir la función $F : \mathbb{N}^2 \longrightarrow \mathbb{N}$ dada por $F(m,n) = \{\pi(m)\}(n) = \mathcal{U}(\pi(m), n)$ y a su vez la función $G : \mathbb{N} \longrightarrow \mathbb{N}$ dada por $G(n) = F(n,n) + 1$.

Ahora podemos concluir que G no es recursiva, pues, si lo fuera, existiría un m tal que $\bigwedge n\, G(n) = F(m,n)$, y en particular $F(n,n)+1 = G(n) = F(n,n)$. A su vez, esto implica que π tampoco puede ser recursiva, porque F se obtiene de $\mathcal{U}$ y π por composición, luego si π fuera recursiva, la función F sería recursiva parcial y a la vez total, luego sería recursiva.

En definitiva, concluimos que una enumeración de los códigos de las funciones recursivas no puede ser recursiva, y en particular esto implica que no existe ningún algoritmo que nos permita determinar si un código de una función recursiva parcial corresponde o no a una función total. ∎

8.4 Funciones demostrablemente recursivas

El último ejemplo de la sección precedente pone de manifiesto que el problema de si un código de una función recursiva parcial corresponde o no a una función recursiva (es decir, si define una función total), no es trivial en absoluto, en el sentido de que no hay ningún algoritmo que lo resuelva. En general, justificar que un código de una función recursiva parcial define una función recursiva requiere un razonamiento específico, y sobre todo razonamiento cabe preguntarse en qué teorías formales puede formalizarse. Esto nos lleva a la definición siguiente:

Definición 8.31 Una función $F : \mathbb{N}^r \longrightarrow \mathbb{N}$ es *demostrablemente recursiva* en una teoría axiomática T sobre $\ulcorner\mathcal{L}_a\urcorner$ si existe una fórmula $\phi \in \mathrm{Form}(\ulcorner\mathcal{L}_a\urcorner)$ de tipo Σ_1 con $r+1$ variables libres tal que

$$\bigwedge s \in \mathbb{N}^r \bigwedge n(F(s) = n \leftrightarrow \mathbb{N} \vDash_{\Sigma_1} \phi[s, n])$$

y además

$$\underset{T}{\vdash} \bigwedge x_0 \cdots x_{r \dot{-} 1} \overset{1}{\bigvee} x_r \, \phi(x_0, \ldots, x_r).$$

Según 8.21, la primera condición equivale a que F es recursiva, mientras que la segunda expresa que el hecho de que F es ciertamente una función total puede probarse en T.

Naturalmente, que una función recursiva F sea o no demostrablemente recursiva en una teoría T depende únicamente de las definiciones por minimización que contenga una definición de F en los términos del teorema 8.28. Para precisar esta idea introducimos el concepto siguiente:

Si $G : \mathbb{N}^{r+1} \longrightarrow \mathbb{N}$ es una función demostrablemente recursiva en una teoría T y la fórmula ϕ cumple la definición anterior, diremos que $F : \mathbb{N}^r \longrightarrow \mathbb{N}$ está definida por *minimización demostrable en* T a partir de G si:

1. $\bigwedge s \in \mathbb{N}^r \bigvee n \, G(s, n) = 0$.
2. $\underset{T}{\vdash} \bigwedge x_1 \cdots x_{r \dot{-} 1} \bigvee x_r \, \phi(x_0, \ldots, x_r, 0)$.
3. $\bigwedge s \in \mathbb{N}^r \, F(s) = \mu n \, G(s, n) = 0$.

Obviamente, en este caso F es simplemente la función definida por minimización a partir de G, pero puede suceder que G defina una función por minimización y que la función definida por minimización a partir de G no esté definida por minimización demostrable en T.

El primer paso para relacionar esto con la demostrabilidad recursiva es el teorema siguiente:

Teorema 8.32 *Sea T una teoría sobre $\mathcal{L}_a$ que extienda a $\mathrm{I}\Sigma_1$. Las funciones recursivas elementales son demostrablemente recursivas en T y toda función definida por composición, recursión o minimización demostrable en T a partir de funciones demostrablemente recursivas en T es demostrablemente recursiva en T.*

Demostración: Las funciones recursivas elementales S, C y P_i^n son, de hecho, demostrablemente recursivas en $K_{\mathcal{L}_a}$, pues las fórmulas $\ulcorner x_1 = x_0' \urcorner$, $\ulcorner x_1 = 0 \urcorner$, $\ulcorner x_n = x_i \urcorner$ satisfacen trivialmente la definición.

En general, se trata de probar que si unas funciones dadas son demostrablemente recursivas y están definidas por unas ciertas fórmulas, entonces las fórmulas consideradas en el teorema 8.4 prueban que la función definida por composición, recursión o minimización demostrable en T a partir de ellas es también demostrablemente recursiva en T. La prueba es rutinaria, así que vamos a detallar únicamente el caso de la minimización demostrable.

La hipótesis es que tenemos una función $G : \mathbb{N}^{r+1} \longrightarrow \mathbb{N}$ demostrablemente recursiva en T definida por la fórmula $\alpha(x_0, \ldots, x_{r+1})$, lo cual significa que

$$\bigwedge s \in \mathbb{N}^{r+1} \bigwedge n(G(s) = n \leftrightarrow \mathbb{N} \vDash_{\Sigma_1} \alpha[s, n])$$

y además

$$\underset{T}{\vdash} \bigwedge x_0 \cdots x_r \overset{1}{\bigvee} x_{r+1}\, \alpha(x_0, \ldots, x_{r+1}).$$

Además suponemos que G define una función $F : \mathbb{N}^r \longrightarrow \mathbb{N}$ por minimización demostrable en T, (en particular, por minimización), luego F está definida por la fórmula dada en 8.4, a saber:

$$\beta(x_0, \ldots, x_r) \equiv \bigwedge i < x_r \bigvee u(\alpha(x_0, \ldots, x_{r\dot{-}1}, i, u) \wedge u \neq 0) \wedge \alpha(x_0, \ldots, x_r, 0),$$

es decir, que se cumple:

$$\bigwedge s \in \mathbb{N}^r \bigwedge n(F(s) = n \leftrightarrow \mathbb{N} \vDash_{\Sigma_1} \beta[s, n]).$$

Falta probar que

$$\underset{T}{\vdash} \bigwedge x_0 \cdots x_{r\dot{-}1} \overset{1}{\bigvee} x_r\, \beta(x_0, \ldots, x_r)$$

Para ello, razonando en T, fijamos $x_0, \ldots, x_{r\dot{-}1}$ y por la condición 2 de la definición de minimización demostrable, sabemos que $\bigvee x_r\, \alpha(x_0, \ldots, x_r, 0)$. Como la teoría T extiende a $\mathrm{I}\Sigma_1$, podemos usar el teorema 4.29, que nos da que

$$\overset{1}{\bigvee} x_r(\bigwedge i < x_r\, \neg\alpha(x_0, \ldots, x_{r\dot{-}1}, i, 0) \wedge \alpha(x_0, \ldots, x_r, 0)).$$

Ahora, fijado x_r que cumpla esto, tomamos $i < x_r$, con lo que sabemos que $\neg\alpha(x_0, \ldots, x_{r\dot{-}1}, i, 0)$. Sin embargo, como G es demostrablemente recursiva, tenemos que existe un único u tal que $\alpha(x_0, \ldots, x_{r\dot{-}1}, i, u)$, por lo que tiene que ser $u \neq 0$, y esto prueba $\beta(x_0, \ldots, x_r)$.

Falta probar la unicidad, pero si suponemos

$$\beta(x_0, \ldots, x_{r\dot{-}1}, y) \wedge \beta(x_0, \ldots, x_{r\dot{-}1}, z),$$

como T extiende a $\mathrm{I}\Sigma_1$, tenemos que $y < z \vee z < y \vee y = z$. Sólo tenemos que descartar los dos primeros casos y, por simetría, consideramos únicamente $y < z$. En este caso, por definición de $\beta(x_0, \ldots, x_{r\dot{-}1}, z)$, tenemos que existe un $u \neq 0$ tal que $\alpha(x_0, \ldots, x_{r\dot{-}1}, z, u)$, mientras que, por definición de $\beta(x_0, \ldots, x_{r\dot{-}1}, z)$, se cumple $\alpha(x_0, \ldots, x_{r\dot{-}1}, z, 0)$. Por último, como α satisface la definición de función demostrablemente recursiva, en T la condición de unicidad nos da que $u = 0$, lo que nos da una contradicción. ∎

De este teorema se sigue inmediatamente que las funciones recursivas primitivas son demostrablemente recursivas en $\mathrm{I}\Sigma_1$. Una demostración formal en ACA_0 requiere razonar por inducción que si $c \in \mathcal{C}_{\mathrm{rp}}$ cumple $\mathrm{Nar}(c) = r$, entonces

$$\underset{\mathrm{I}\Sigma_1}{\vdash} \bigwedge x_0 \cdots x_{r\dot{-}1} \overset{1}{\bigvee} x_r \mathcal{F}(c).$$

No es trivial que el recíproco también es cierto:

Teorema 8.33 *Las funciones demostrablemente recursivas en* $\mathrm{I}\Sigma_1$ *son las funciones recursivas primitivas.*

DEMOSTRACIÓN: Ya hemos probado una implicación. Si $F : \mathbb{N}^r \longrightarrow \mathbb{N}$ es una función demostrablemente recursiva en $\mathrm{I}\Sigma_1$, sea $\phi \in \mathrm{Form}(\ulcorner\mathcal{L}_a\urcorner)$ la fórmula que lo justifica, de modo que

$$\underset{\mathrm{I}\Sigma_1}{\vdash} \bigwedge x_0 \cdots x_{r\dot{-}1} \overset{1}{\bigvee} x_r\, \phi(x_0, \ldots, x_r).$$

El teorema [CS 1.23] implica que existe un funtor f de $\ulcorner\mathcal{L}_{\mathrm{arp}}\urcorner$ tal que

$$\underset{\mathrm{ARP}}{\vdash} \bigwedge x_0 \cdots x_{r\dot{-}1} \phi(x_0, \ldots, x_{r\dot{-}1}, f(x_0, \ldots, x_{r\dot{-}1}))$$

o, equivalentemente,

$$\underset{\mathrm{ARP}}{\vdash} \bigwedge x_0 \cdots x_{r\dot{-}1} \bigvee u(f(x_0, \ldots, x_{r\dot{-}1}) = u \wedge \phi(x_0, \ldots, x_{r\dot{-}1}, u))$$

Sea G la función recursiva primitiva asociada al funtor f por el teorema 8.15. Si interpretamos la fórmula $f(x_0, \ldots, x_{r\dot{-}1}) = u$ como su traducción a $\ulcorner\mathcal{L}_a\urcorner$ según el teorema 4.49, la fórmula anterior puede probarse en $\mathrm{I}\Sigma_1$, luego el teorema [CS 1.25] nos da que, para todo $s \in \mathbb{N}^r$,

$$\mathbb{N} \vDash_{\Sigma_1} \bigvee u(f(x_0, \ldots, x_{r\dot{-}1}) = u \wedge \phi(x_0, \ldots, x_{r\dot{-}1}, u))[s],$$

luego existe un n tal que

$$\mathbb{N} \vDash_{\Sigma_1} (f(x_0, \ldots, x_{r\dot{-}1}) = u)[s, n], \quad \mathbb{N} \vDash_{\Sigma_1} \phi(x_0, \ldots, x_{r\dot{-}1}, u)[s, n].$$

Por el teorema 8.15 la primera fórmula equivale a $G(s) = n$, mientras que la segunda equivale a que $F(s) = n$, luego tenemos que $F = G$ es recursiva primitiva. ∎

Ahora podemos caracterizar las funciones demostrablemente recursivas:

Teorema 8.34 *Una función* $F : \mathbb{N}^r \longrightarrow \mathbb{N}$ *es demostrablemente recursiva en una teoría* T *si y sólo si existe un* k *con* $\ell(k) = m + 1$ *y* $\bigwedge i \leq m\, k_i \geq 1$ *y existe una sucesión de funciones* H*, de modo que* $\bigwedge i \leq r H_i : \mathbb{N}^{k_i} \longrightarrow \mathbb{N}$*,* $H_m = F$ *y cada* H_i *es recursiva elemental, o bien está definida a partir de funciones anteriores por composición, recursión o minimización demostrable en* T*.*

DEMOSTRACIÓN: Supuesto que existe la sucesión H, el teorema 8.32 permite probar por inducción que cada función H_i es demostrablemente recursiva, con lo que F lo es. (Notemos que la fórmula de la inducción es aritmética, pues H es un parámetro fijo en ella.)

Supuesto que F sea demostrablemente recursiva, tomamos una fórmula que la defina, que será de la forma $\bigvee u\, \phi(x_0, \ldots, x_r, u)$, donde ϕ es de tipo Δ_0. Tenemos que

$$\bigwedge s \in \mathbb{N}^r \bigwedge n(F(s) = n \leftrightarrow \mathbb{N} \vDash_{\Sigma_1} \bigvee u\, \phi[s, n])$$

y además

$$\vdash_T \bigwedge x_0 \cdots x_{r\dot{-}1} \overset{1}{\bigvee} x_r \bigvee x_{r+1}\, \phi(x_0, \ldots, x_{r+1}).$$

En particular

$$\vdash_T \bigwedge x_0 \cdots x_{r\dot{-}1} \bigvee u \bigvee vw \leq u(u = \langle v, w\rangle_2 \wedge \phi(x_0, \ldots, x_{r\dot{-}1}, v, w)).$$

La fórmula $\bigvee vw \leq x_r(x_r = \langle v, w\rangle_2 \wedge \phi(x_0, \ldots, x_{r\dot{-}1}, v, w))$ es Δ_0, luego por el teorema 8.19 define una relación recursiva primitiva. Esto significa que la función $G : \mathbb{N}^{r+1} \longrightarrow \mathbb{N}$ dada por

$$G_0(t) = \begin{cases} 1 & \text{si } \mathbb{N} \vDash_{\Sigma_1} \bigvee vw \leq x_r(x_r = \langle v, w\rangle_2 \wedge \phi(x_0, \ldots, x_{r\dot{-}1}, v, w))[t], \\ 0 & \text{si } \neg\mathbb{N} \vDash_{\Sigma_1} \bigvee vw \leq x_r(x_r = \langle v, w\rangle_2 \wedge \phi(x_0, \ldots, x_{r\dot{-}1}, v, w))[t] \end{cases}$$

es recursiva primitiva, y también lo es $G_1(t) = 1 \dot{-} G_0(t)$, que está definida por la fórmula

$$\psi(x_0, \ldots, x_{r+1}) \equiv (\bigvee vw \leq x_r(x_r = \langle v, w\rangle_2 \wedge \phi(x_0, \ldots, x_{r\dot{-}1}, v, w)) \wedge x_{r+1} = 0)$$

$$\vee\, (\neg\bigvee vw \leq x_r(x_r = \langle v, w\rangle_2 \wedge \phi(x_0, \ldots, x_{r\dot{-}1}, v, w)) \wedge x_{r+1} = 1).$$

Además $\vdash_T \bigwedge x_0 \cdots x_{r\dot{-}1} \bigvee x_r \psi(x_0, \ldots, x_r, 0)$, luego concluimos que G_1 define una función $G_2 : \mathbb{N}^r \longrightarrow \mathbb{N}$ por minimización demostrable en T, de modo que

$$G_2(s) = \mu n\, \mathbb{N} \vDash_{\Sigma_1} \bigvee vw \leq x_r(u = \langle v, w\rangle_2 \wedge \phi(x_0, \ldots, x_{r\dot{-}1}, v, w))[s, n].$$

Más explícitamente, para cada $s \in \mathbb{N}^r$ existen n, m tales que $G_2(s) = \langle n, m\rangle_2$ y

$$\mathbb{N} \vDash_{\Sigma_1} \phi(x_0, \ldots, x_{r+1})[s, n, m].$$

En particular,

$$\mathbb{N} \vDash_{\Sigma_1} \bigvee u\, \phi(x_0, \ldots, x_r, u)[s, n],$$

luego $n = F(s)$. Por lo tanto, F es la composición de G_2 con la función recursiva primitiva $G_3(n) = n_1$ que asigna a cada par ordenado su primera componente.

Recapitulando: La función G_1 es recursiva primitiva, luego existe una sucesión de funciones $H_0, \ldots, H_m$ tal que $H_m = G_1$ y cada H_i es recursiva elemental o está definida a partir de funciones anteriores por composición o recursión. A su vez, H_m define una función $H_{m+1} = G_2$ por minimización demostrable en T, y la composición de ésta con $H_{m+2} = G_3$ es la función $H_{m+3} = F$. ∎

8.5 Conjuntos recursivos

Informalmente, un conjunto recursivo es lo mismo que una relación monádica recursiva. Ambos conceptos consisten en un criterio para determinar si un número natural cumple o no algo (llámese pertenecer al conjunto o satisfacer la relación). Sin embargo, los convenios que hemos usado para formalizar las sucesiones finitas de números naturales marcan una diferencia técnica: una relación monádica recursiva es un subconjunto $R \subset \mathbb{N}^1$, y $\mathbb{N}^1$ no es lo mismo que $\mathbb{N}$.

En la práctica, podemos identificar cada función $F : \mathbb{N} \longrightarrow \mathbb{N}$ con la función $F^1 : \mathbb{N}^1 \longrightarrow \mathbb{N}$ dada por $F^1(\langle n\rangle) = F(n)$ y, recíprocamente, toda función $F^1 : \mathbb{N}^1 \longrightarrow \mathbb{N}$ se corresponde de este modo con una función $F : \mathbb{N} \longrightarrow \mathbb{N}$.

De hecho, cuando decimos que una función $F^1 : \mathbb{N}^1 \longrightarrow \mathbb{N}$ es recursiva, lo que queremos decir —de acuerdo con las definiciones que hemos dado— es que existe un algoritmo al que, si le damos un número n (no $\langle n\rangle$), nos calcula $F^1(\langle n\rangle)$, luego lo que estamos diciendo en realidad es que la función $F : \mathbb{N} \longrightarrow \mathbb{N}$ es calculable algorítmicamente.

En la práctica podemos identificar cada funcióN F con la F^1 correspondiente, y los conjuntos recursivos con las relaciones monádicas recursivas, pero vamos a dar aquí definiciones consistentes con los convenios que hemos adoptado y mostraremos que, en efecto, este tipo de identificaciones no dan lugar a ninguna ambigüedad:

Definición 8.35 Un *conjunto recursivo* es un conjunto X tal que la función $\chi^1_X : \mathbb{N}^1 \longrightarrow \mathbb{N}$ dada por

$$\chi^1_X(n) = \begin{cases} 1 & \text{si } n \in X, \\ 0 & \text{si } n \notin X \end{cases}$$

es recursiva, donde adoptamos el convenio usual para funciones r-ádicas (en este caso, con $r = 1$), según el cual $\chi^1_X(n)$ es, en realidad, $\chi^1_X(\langle n\rangle)$.

En la práctica identificaremos χ^1_X con la función característica $\chi_X : \mathbb{N} \longrightarrow \mathbb{N}$ y así podemos parafrasear así la definición:

> Un conjunto es *recursivo* si y sólo si su función característica es recursiva.

Diremos que un conjunto X es Σ_1 si existe una fórmula $\phi(x) \in \mathrm{Form}(\ulcorner\mathcal{L}_a\urcorner)$ con una única variable libre y de tipo Σ_1 tal que

$$\bigwedge n(n \in X \leftrightarrow \mathbb{N} \vDash_{\Sigma_1} \phi[n]),$$

donde la parte derecha ha de entenderse como $\mathbb{N} \vDash_{\Sigma_1} \phi[s]$ con $s = \langle n\rangle$. Análogamente se define un conjunto Π_1, y un conjunto es Δ_1 si es a la vez Σ_1 y Π_1.

Ahora bien, es inmediato que un conjunto X es Δ_1 si y sólo si lo es la relación monádica dada por $R_X(n) \leftrightarrow n \in X$ (donde $R_X(n)$ ha de entenderse como $\langle n\rangle \in R \subset \mathbb{N}^1$), lo cual equivale, por 8.20 a que la relación R_X sea recursiva, que a su vez equivale a que la función χ^1_X (que es precisamente la función característica de R_X) sea recursiva, y esto a su vez equivale a que X sea recursivo. En conclusión:

Teorema 8.36 *Un conjunto es recursivo si y sólo si es Δ_1.*

Si $\alpha(x, x_1, \ldots, x_n)$ es una fórmula de $\mathcal{L}_a$ de tipo Σ_1 o Π_1, entonces el conjunto

$$X \equiv \{x \mid \alpha(x, x_1, \ldots, x_n)\}$$

es de tipo Σ_1 o Π_1, pues está definido por la fórmula

$$\phi \equiv \ulcorner\alpha\urcorner(x, 0^{(x_1)}, \ldots, 0^{(x_n)}) \in \mathrm{Form}(\ulcorner\mathcal{L}_a\urcorner).$$

Recíprocamente, todo conjunto de tipo Σ_1 o Π_1 es de esta forma, consideramos la fórmula

$$\alpha(n) \equiv \mathbb{N} \vDash_{\Sigma_1} \phi[n] \quad \text{o} \quad \alpha(n) \equiv \mathbb{N} \vDash_{\Pi_1} \phi[n].$$

Ahora podemos interpretar el axioma de comprensión recursiva como el axioma que expresa la existencia de todos los conjuntos recursivos (es decir, que, para todo conjunto recursivo, el axioma afirma la existencia de un objeto de segundo orden cuyos elementos son precisamente los de dicho conjunto). Puesto que es el único axioma de ACR_0 que postula la existencia de conjuntos, podemos afirmar que en ACR_0 no se puede demostrar la existencia de ningún conjunto que no sea recursivo (pero una cosa es que un conjunto sea recursivo y otra que su existencia pueda demostrarse en ACR_0).

Obviamente, un conjunto X es Π_1 si y sólo si su complementario $\mathbb{N}\setminus X$ es Σ_1, luego un conjunto X es recursivo si y sólo si tanto X como $\mathbb{N}\setminus X$ son Σ_1.

Por ello, a los conjuntos Σ_1 se los llama también conjuntos *semirrecursivos,* pues ser Σ_1 es la mitad de lo que tiene que cumplir un conjunto para ser recursivo.

Los conjuntos semirrecursivos se llaman también conjuntos *recursivamente numerables,* debido al teorema siguiente:

Teorema 8.37 *Un conjunto no vacío es Σ_1 si y sólo si es el rango de una función recursiva: $F : \mathbb{N} \longrightarrow \mathbb{N}$.*

Demostración: Si X es un conjunto Σ_1, existe una fórmula $\alpha(x_0, x_1)$ de tipo Δ_0 tal que

$$n \in X \leftrightarrow \bigvee m\, \mathbb{N} \vDash_{\Sigma_1} \alpha[m, n].$$

Tomamos $a \in X$ y definimos

$$F = \{\langle k, l\rangle_2 \mid \bigvee m(k = \langle m, n\rangle_2 \wedge (\mathbb{N} \vDash_{\Sigma_1} \alpha[m, n] \wedge l = n)$$

$$\vee \neg\mathbb{N} \vDash_{\Pi_1} \alpha[m, n] \wedge l = a)\}.$$

Así $F : \mathbb{N} \longrightarrow \mathbb{N}$ es una función Σ_1, luego recursiva, y su rango es claramente X.

Recíprocamente, si $F : \mathbb{N} \longrightarrow \mathbb{N}$ es una función recursiva con rango X, entonces

$$n \in X \leftrightarrow \bigvee m\ \langle m, n\rangle_2 \in G(F),$$

y la gráfica $G(F)$ es un conjunto Σ_1, de donde se sigue que la fórmula de la derecha es Σ_1 y X también. ∎

Así pues, si un conjunto es semirrecursivo, podemos ir enumerando sus elementos, de modo que si un número natural está en el conjunto, tarde o temprano lo sabremos, pues aparecerá en la sucesión, pero si no pertenece, nunca lo llegaremos a saber, pues siempre nos quedará la duda de si no aparecerá más adelante.

Teorema 8.38 *Existe un conjunto Σ_1 universal U, es decir, un conjunto Σ_1 tal que para todo conjunto X de tipo Σ_1 existe un número m tal que,*

$$\bigwedge n(n \in X \leftrightarrow \langle m, n\rangle_2 \in U).$$

Demostración: Sea $\phi(\alpha)$ la fórmula de $\mathcal{L}_a$ que afirma que $\alpha \in \mathrm{Form}(\ulcorner\mathcal{L}_a\urcorner)$ es una fórmula de tipo Σ_1 con x_0 como única variable libre. Claramente es una fórmula de tipo Σ_1, y nos permite definir

$$U \equiv \{\langle \alpha, n\rangle \mid \phi(\alpha) \wedge \mathbb{N} \vDash_{\Sigma_1} \alpha[n]\},$$

que es un conjunto de tipo Σ_1. Si X es cualquier conjunto de tipo Σ_1, existe α tal que $\phi(\alpha)$ y

$$\bigwedge n(n \in X \leftrightarrow \mathbb{N} \vDash_{\Sigma_1} \alpha[n]),$$

y esto equivale a la condición del enunciado con $m = \alpha$. ∎

Como consecuencia:

Teorema 8.39 *Un conjunto semirrecursivo universal es un ejemplo de conjunto semirrecursivo no recursivo.*

Demostración: Sea U un conjunto semirrecursivo universal. Si fuera recursivo, también lo sería $\bar{U} = \mathbb{N} \setminus U$, y el conjunto

$$X \equiv \{n \mid \langle n, n\rangle_2 \in \bar{U}\}$$

sería Σ_1 (de hecho sería recursivo, pero nos basta con que sería Σ_1), pues la fórmula que lo define es Σ_1. Por la universalidad de U, existiría un m tal que

$$\bigwedge n(n \in X \leftrightarrow \langle m, n\rangle_2 \in U),$$

luego en particular $\langle m, m\rangle_2 \in U \leftrightarrow m \in X \leftrightarrow \langle m, m\rangle_2 \in \bar{U} \leftrightarrow \langle m, m\rangle \notin U$, y tenemos una contradicción. ∎

A partir de este ejemplo podemos encontrar otro más sustancioso:

Teorema 8.40 *El conjunto de todas las sentencias $\alpha \in \mathrm{Form}(\ulcorner\mathcal{L}_a\urcorner)$ de tipo Σ_1 que cumplen $\mathbb{N} \vDash_{\Sigma_1} \alpha$ es semirrecursivo, pero no recursivo.*

Demostración: Sea X el conjunto indicado, que es semirrecursivo porque la fórmula que lo define es Σ_1. Supongamos que fuera recursivo, es decir, Δ_1, de modo que existe una fórmula $\beta \in \mathrm{Form}(\ulcorner\mathcal{L}_a\urcorner)$ de tipo Π_1 tal que

$$\bigwedge m(m \in X \leftrightarrow \mathbb{N} \vDash_{\Pi_1} \beta(m)).$$

Tomemos un U que sea semirrecursivo, pero no recursivo. Como es Σ_1, existe una fórmula $\alpha(x)$ de tipo Σ_1 tal que, para todo número natural n, se cumple

$$n \in U \leftrightarrow \mathbb{N} \vDash_{\Sigma_1} \alpha[n] \leftrightarrow \mathbb{N} \vDash_{\Sigma_1} \alpha(0^{(n)}) \leftrightarrow \alpha(0^{(n)}) \in X \leftrightarrow$$

$$\bigwedge m(m = \alpha(0^{(n)}) \to m \in X) \leftrightarrow \bigwedge m(m = \alpha(0^{(n)}) \to \mathbb{N} \vDash_{\Pi_1} \beta[m]),$$

y la última fórmula es Π_1, luego llegamos a que U es Δ_1, es decir, recursivo, lo cual es falso. ∎

Así pues, no existe ningún algoritmo que nos permita determinar si una sentencia de $\mathcal{L}_a$ de tipo Σ_1 es verdadera o falsa. Podremos razonarlo en muchos casos particulares, pero no existe un algoritmo general que resuelva el problema.

Refinando esta idea podemos obtener una profunda limitación sobre la capacidad de razonamiento formal:

Consideremos una teoría axiomática cualquiera T sobre un lenguaje formal cualquiera $\mathcal{L}$. Sólo vamos a exigir que $\mathcal{L}$ sea recursivo y que T sea semirrecursiva en el sentido de la definición 6.30, lo que técnicamente significa que la fórmula

$$\Gamma \underset{T}{\overset{d}{\vdash}} \alpha$$

es Σ_1, y se interpreta como que podemos enumerar explícitamente los axiomas (y, consecuentemente, los teoremas) de T. Suponemos además que T interpreta a $K_{\ulcorner\mathcal{L}_a\urcorner}$ en el sentido de la definición 5.19, es decir, que cada fórmula $\alpha \in \mathrm{Form}(\ulcorner\mathcal{L}_a\urcorner)$ tiene una traducción $\bar{\alpha} \in \mathrm{Form}(\mathcal{L})$. Supondremos también que la interpretación es semirrecursiva en el sentido de que la fórmula $\beta = \bar{\alpha}$ es Σ_1.

Estas hipótesis son triviales, en el sentido de que sólo estamos suponiendo algo más débil que el hecho de que sabemos reconocer si una fórmula de $\mathcal{L}$ es o no un axioma de T y que sabemos traducir cada fórmula de $\ulcorner\mathcal{L}_a\urcorner$ a una fórmula de $\mathcal{L}$. Decimos más débil porque esto sería así si supusiéramos que T y la interpretación son recursivas, cuando sólo necesitamos suponer que son semirrecursivas.

En estas condiciones podemos demostrar:

Teorema 8.41 *Sea T una teoría semirrecursiva que interprete a $K_{\ulcorner\mathcal{L}_a\urcorner}$ con una interpretación recursiva, es decir, de modo que las fórmulas $\alpha \in \mathrm{Ax}(T)$ y $\beta = \bar{\alpha}$ son Σ_1. Entonces se tiene que dar uno de los dos casos siguientes:*

1. *Existe una sentencia $\alpha \in \mathrm{Form}\ulcorner\mathcal{L}_a\urcorner$ de tipo Σ_1 o Π_1 que es falsa (es decir, tal que $\neg\mathbb{N} \vDash_{\Sigma_1} \alpha$ o $\neg\mathbb{N} \vDash_{\Pi_1} \alpha$) y $\underset{T}{\vdash} \bar{\alpha}$.*

2. *Existe una sentencia $\alpha \in \mathrm{Form}\ulcorner\mathcal{L}_a\urcorner$ de tipo Σ_1 o Π_1 que es verdadera (cumple $\mathbb{N} \vDash_{\Sigma_1} \alpha$ o $\mathbb{N} \vDash_{\Pi_1} \alpha$) y $\neg\underset{T}{\vdash} \bar{\alpha}$.*

En otras palabras: o bien la teoría T permite demostrar fórmulas falsas, o bien existe una fórmula verdadera que no se puede demostrar en T.

Demostración: Por el teorema de Craig 6.31 podemos suponer que T es, de hecho, recursiva, lo cual significa que la fórmula

$$\Gamma \underset{T}{\overset{d}{\vdash}} \alpha$$

es Δ_1 (en $\mathrm{I}\Sigma_1$, luego en la teoría ACA_0 en la que estamos razonando).

Supongamos que T no permite demostrar (traducciones de) sentencias falsas de tipo Σ_1 o Π_1. En particular esto implica que T es consistente. Y supongamos también (por reducción al absurdo), que si α es una sentencia verdadera de tipo Σ_1 o Π_1, entonces $\underset{T}{\vdash} \bar{\alpha}$.

Así, si α es una sentencia de tipo Σ_1, o bien $\mathbb{N} \vDash_{\Sigma_1} \alpha$, en cuyo caso $\underset{T}{\vdash} \bar{\alpha}$, o bien $\mathbb{N} \vDash_{\Pi_1} \neg\alpha$, en cuyo caso $\underset{T}{\vdash} \neg\bar{\alpha}$. En cualquier caso

$$\bigvee d \, (\overset{d}{\underset{T}{\vdash}} \, \bar{\alpha} \vee \overset{d}{\underset{T}{\vdash}} \, \neg\bar{\alpha}).$$

Tomemos un conjunto U que sea semirrecursivo, pero no recursivo. Entonces existe una fórmula $\phi(x)$ de $\ulcorner\mathcal{L}_a\urcorner$ de tipo Σ_1 con x como única variable libre tal que

$$\bigwedge n(n \in U \leftrightarrow \mathbb{N} \vDash_{\Sigma_1} \phi(0^{(n)}).$$

Aplicando lo anterior a las sentencias $\phi(0^{(n)})$ de $\ulcorner\mathcal{L}_a\urcorner$, tenemos que

$$\bigwedge n \bigvee d \, (\overset{d}{\underset{T}{\vdash}} \, \overline{\phi(0^{(n)})} \vee \overset{d}{\underset{T}{\vdash}} \, \neg\overline{\phi(0^{(n)})} \,).$$

La fórmula $\psi(n, d)$ que hay tras $\bigwedge n \bigvee d$ es Δ_1, luego la fórmula

$$\chi(n, d) \equiv \psi(n, d) \wedge \bigwedge u < d \, \neg\psi(n, d)$$

también lo es, y cumple que $\bigwedge n \overset{1}{\bigvee} d \, \chi(n, d)$, luego la función $F : \mathbb{N} \longrightarrow \mathbb{N}$ dada por

$$F \equiv \{\langle n, d\rangle_2 \mid \chi(n, d)\}$$

es Σ_1, luego, según el teorema 8.21, es una función recursiva y, de hecho, la fórmula $F(n) = d$ es Δ_1. La función F asigna a cada n la menor demostración en T de la traducción de $\phi(0^{(n)})$ o de su negación. Consideramos el conjunto

$$U^* \equiv \{n \mid \bigwedge d(F(n) = d \to \overset{d}{\underset{T}{\vdash}} \, \overline{\phi(0^{(n)})})\},$$

que es de tipo Π_1. Vamos a ver que $U = U^*$, con lo que tendremos una contradicción, pues esto significa que U es recursivo. En efecto, si $n \in U$, entonces $\mathbb{N} \vDash_{\Sigma_1} \phi(0^{(n)})$, luego, por hipótesis, $\underset{T}{\vdash} \overline{\phi(0^{(n)})}$ y, como T es consistente, no puede ser que $d = F(n)$ demuestre lo contrario, luego si $F(n) = d$ se cumple $\overset{d}{\underset{T}{\vdash}} \, \overline{\phi(0^{(n)})}$, luego $n \in U^*$.

Recíprocamente, si $n \in U^*$, tomando $d = F(n)$ tenemos que $\overset{d}{\underset{T}{\vdash}} \, \overline{\phi(0^{(n)})}$, pero por hipótesis en T no se demuestran traducciones de sentencias falsas, luego $\mathbb{N} \vDash_{\Sigma_1} \phi(0^{(n)})$, luego $n \in U$. ∎

Así pues, es imposible construir una teoría axiomática razonable que nos permita demostrar todas las afirmaciones verdaderas sobre los números naturales. "Razonable" significa que le estamos pidiendo que podamos identificar en la práctica cuáles son sus axiomas y que sepamos qué fórmula del lenguaje de la teoría expresa cada sentencia de $\ulcorner\mathcal{L}_a\urcorner$ de tipo Σ_1 o Π_1. Sin estas hipótesis, nos bastaría tomar como axiomas de T todas las sentencias de tipo Σ_1 o Π_1 que sean verdaderas y tendríamos una teoría en la que se pueden demostrar ¡en una

línea! todas las sentencias verdaderas de tipo Σ_1 o Π_1 (y ninguna falsa[5]), pero el problema es que no sabríamos qué sentencias son axiomas y cuáles no, con lo que no podríamos distinguir si una presunta demostración es válida o no.

Observemos que si exigimos que la teoría T no sólo interprete a $K_{\mathcal{L}_a}$, sino a la aritmética de Robinson Q, entonces el teorema 6.17 nos asegura que (la traducción de) toda fórmula verdadera de tipo Σ_1 es demostrable en T, luego la conclusión del segundo caso del teorema anterior es, más concretamente, que existe una fórmula de tipo Π_1 verdadera y no demostrable en T.

Nota Del teorema anterior se deduce una consecuencia que no es formalizable en ACA_0:

> *El conjunto de las sentencias α de $\mathcal{L}_a$ tales que $\mathbb{N} \models \alpha$ no es semirecursivo.*

En otras palabras, no existe ningún algoritmo que permita enumerar todas las sentencias aritméticas verdaderas (y mucho menos determinar si una sentencia aritmética dada es verdadera o falsa).

En efecto, si el conjunto de las sentencias verdaderas de $\mathcal{L}_a$ fuera semirrecursivo, la teoría axiomática T que tiene a dichas sentencias por axiomas debería cumplir uno de los dos casos contemplados en el teorema anterior, pero ambos son obviamente imposibles (no puede haber una sentencia falsa demostrable, pues todas las consecuencias de sentencias verdaderas tienen que ser verdaderas, ni puede haber una sentencia verdadera no demostrable, pues todas las sentencias verdaderas son axiomas). ∎

En el capítulo siguiente profundizaremos en los resultados de este tipo.

[5]Que no se puede demostrar ninguna sentencia falsa de tipo Σ_1 o Π_1 es consecuencia del teorema 6.20, pues si α es una sentencia demostrable en T a partir de unos axiomas $\alpha_1, \ldots, \alpha_n$, entonces la fórmula $\alpha_1 \wedge \cdots \wedge \alpha_n \to \alpha$ es un teorema lógico de tipo Σ_2, luego 6.20 nos da que $\mathbb{N} \vDash_{\Sigma_2} (\alpha_1 \wedge \cdots \wedge \alpha_n \to \alpha)$, y esto implica que

$$\neg\mathbb{N} \vDash_{\Sigma_2} \alpha_1 \vee \cdots \neg\mathbb{N} \vDash_{\Sigma_2} \alpha_n \vee \neg\mathbb{N} \vDash_{\Sigma_2} \alpha,$$

pero, como se cumple $\mathbb{N} \vDash_{\Sigma_1} \alpha_i$ o $\mathbb{N} \vDash_{\Pi_1} \alpha_i$, también $\mathbb{N} \vDash_{\Sigma_2} \alpha_i$, luego tiene que ser $\mathbb{N} \vDash_{\Sigma_2} \alpha$, y esto implica $\mathbb{N} \vDash_{\Sigma_1} \alpha$ o $\mathbb{N} \vDash_{\Pi_1} \alpha$.

Capítulo IX

Incompletitud

En el capítulo anterior hemos probado el teorema 8.41, que pone en evidencia serias limitaciones al estudio de los números naturales: en cualquier teoría axiomática recursiva (es decir, en la que sepamos distinguir qué es un axioma), si no es posible demostrar afirmaciones falsas sobre números naturales, entonces hay afirmaciones (verdaderas) de tipo Π_1 que no son demostrables. Las afirmaciones de tipo Π_1 son las más sencillas para las que esto puede suceder: son de la forma $\bigwedge n\,\alpha(n)$, donde $\alpha(n)$ es una fórmula Δ_0, lo que significa que sabemos comprobar si cualquier número natural n cumple o no $\alpha(n)$, pero —según vemos— hay casos en los que es imposible demostrar que todos los números naturales cumplen $\alpha(n)$, a pesar de que es así.

Esto es lo que afirma el conocido como primer teorema de incompletitud de Gödel, pero en este capítulo veremos que podemos demostrarlo en $\mathrm{I}\Sigma_1$ (mientras que en el capítulo anterior lo hemos demostrado en ACA_0), y a partir de él probaremos el segundo teorema de incompletitud, que es un resultado fundamental de la lógica matemática del que ya hemos constatado muchas evidencias que permitirían conjeturarlo si no supiéramos cómo demostrarlo. En efecto, en 6.27 hemos demostrado la consistencia de la aritmética de Robinson Q, pero no la hemos demostrado en Q, sino en $\mathrm{I}\Sigma_1$. Similarmente, en 6.22 hemos visto que la consistencia de $\mathrm{I}\Sigma_1$ puede probarse en $\mathrm{I}\Sigma_2$, y la de $\mathrm{I}\Sigma_2$ en $\mathrm{I}\Sigma_3$, etc., pero si el lector intenta refinar los argumentos para probar la consistencia de $\mathrm{I}\Sigma_n$ en el propio $\mathrm{I}\Sigma_n$, verá que todos sus intentos fracasan, porque eso es precisamente lo que afirma el segundo teorema de incompletitud: que cualquier teoría "razonable" es incapaz de demostrar su propia consistencia. Así, por ejemplo, podemos afirmar que la aritmética de Peano AP es consistente, porque tiene como modelo el conjunto $\mathbb{N}$ de los números naturales con las interpretaciones obvias de todos los signos de $\mathcal{L}_a$, pero la prueba formal que hemos dado tras el teorema 7.45 no es formalizable en AP, sino en RA_0. Precisamente, el segundo teorema de incompletitud nos asegura que es imposible demostrar Consis AP en la propia aritmética de Peano AP, luego tampoco es posible demostrarla en ACA_0 (ya que esta teoría prueba las mismas fórmulas de $\mathcal{L}_a$ que AP). En particular, en ACA_0 no puede demostrarse la existencia de un modelo de AP, pues ello implicaría la

consistencia de AP. Esto implica que el principio de recursión aritmética que determina la teoría AR_0 no es demostrable en ACA_0, es decir, que AR_0 es una teoría más fuerte que ACA_0.

Éstas son sólo algunas consecuencias de los teoremas de incompletitud de Gödel, pero vamos a ver algunas más.

9.1 El primer teorema de incompletitud

Como en el capítulo VI, vamos a razonar en el contexto siguiente:

1. Consideramos como metateoría $\mathrm{I}\Sigma_1$, sobre un lenguaje formal $\mathcal{L}_a^0$ definido informalmente.

2. En esta metateoría consideramos el lenguaje formal $\mathcal{L}_a$ y las teorías axiomáticas Q e $\mathrm{I}\Sigma_1$, así como una teoría arbitraria T definida sobre un lenguaje formal $\mathcal{L}$ que interpreta a Q. Esto significa (definición 5.19) que cada fórmula α de $\mathcal{L}_a$ tiene una traducción $\bar{\alpha}$ a $\mathcal{L}$, de modo que los axiomas de Q son teoremas de T. Llamaremos *fórmulas aritméticas* de $\mathcal{L}$ a las traducciones de fórmulas de $\mathcal{L}_a$.

   Sobrentendemos que el lenguaje $\mathcal{L}$ es recursivo, es decir, que está definido en la metateoría mediante fórmulas de tipo Δ_1.

3. A su vez, en la teoría $\mathrm{I}\Sigma_1$ podemos definir las teorías $\ulcorner\mathrm{I}\Sigma_1\urcorner$ y $\ulcorner T\urcorner$ que formalizan las definiciones de las teorías respectivas en la metateoría $\mathrm{I}\Sigma_1$.

Los resultados de este capítulo son consecuencias del teorema siguiente, que nos permite construir fórmulas que "hablen de sí mismas":

Teorema 9.1 *Sea T una teoría axiomática sobre un lenguaje $\mathcal{L}$ que interprete a Q y sea $\psi(y)$ una fórmula (aritmética) de $\mathcal{L}$ con y como única variable libre. Entonces existe una sentencia (aritmética) ϕ de $\mathcal{L}$ tal que*

$$\vdash_T (\phi \leftrightarrow \psi(\ulcorner\phi\urcorner)).$$

Grosso modo, el teorema afirma que, para toda propiedad ψ, existe una sentencia ϕ que equivale a "yo cumplo la propiedad ψ", entendiendo que "yo" es el número natural $\ulcorner\phi\urcorner$ que formaliza a ϕ en $\mathcal{L}$ (o, más precisamente, el numeral de $\mathcal{L}$ correspondiente al numeral de $\mathcal{L}_a$ que formaliza a ϕ en $\mathrm{I}\Sigma_1$, de modo que en $\mathrm{I}\Sigma_1$ se prueba que $\ulcorner\phi\urcorner \in \mathrm{Form}(\ulcorner\mathcal{L}\urcorner)$).

Demostración: Consideramos la fórmula (de $\mathcal{L}_a^0$):

$$\sigma_0(x,y) \equiv x \in \mathrm{Form}(\mathcal{L}) \wedge y = \mathtt{S}_u^{0^{(x)}} x,$$

donde $u \equiv \ulcorner x\urcorner \in \mathrm{Var}(\mathcal{L}_a)$ es la variable de $\mathcal{L}_a$ que formaliza a la variable x de $\mathcal{L}_a^0$ (que es un numeral de $\mathcal{L}_a^0$).

El hecho de que el lenguaje $\mathcal{L}$ sea recursivo implica que σ_0 es una fórmula Δ_1. Si $\alpha(u) \in \mathrm{Form}(\mathcal{L})$, el único y que cumple $\sigma_0(\alpha, y)$ es $y = \mathtt{S}_u^{0^{(\alpha)}}\alpha \equiv \alpha(\ulcorner\alpha\urcorner)$. Sustituyendo σ_0 por una fórmula equivalente, podemos suponer que cumple el teorema 6.29 con $F(\alpha) = \ulcorner\alpha\urcorner$, es decir, que es una fórmula de tipo Σ_1 tal que, llamando $\sigma \equiv \ulcorner\sigma_0\urcorner \in \mathrm{Form}(\mathcal{L}_a)$, para toda $\alpha(u) \in \mathrm{Form}(\mathcal{L})$, se cumple

$$\underset{\mathrm{Q}}{\vdash} \bigwedge v(\sigma(\ulcorner\alpha\urcorner, v) \leftrightarrow v = \ulcorner\alpha(\ulcorner\alpha\urcorner)\urcorner).$$

Llamamos $\bar{\sigma} \in \mathrm{Form}(\mathcal{L})$ a la traducción de σ a $\mathcal{L}$ y consideramos la fórmula

$$\delta(u) \equiv \bigvee v \in \mathbb{N}(\bar{\sigma}(u, v) \wedge \psi(v)) \in \mathrm{Form}(\mathcal{L}).$$

Notemos que en la metateoría se demuestra que $\delta(u)$ es una fórmula con u como única variable libre, pero esto no contradice que "δ" sea un numeral concreto de $\mathcal{L}_a^0$, y a su vez, su formalización $\ulcorner\delta\urcorner \equiv 0^{(\delta)}$ es un numeral de $\mathcal{L}_a$.

Observemos además que si la fórmula ψ es aritmética, es decir, si es la traducción de una fórmula ψ_0 de $\mathcal{L}_a$, entonces $\delta(u)$ también es una fórmula aritmética, la traducción de la fórmula $\bigvee v(\sigma(u, v) \wedge \psi_0(v))$ de $\mathcal{L}_a$. Llamamos

$$\phi \equiv \delta(\ulcorner\delta\urcorner) \equiv \bigvee v \in \mathbb{N}(\bar{\sigma}(\ulcorner\delta\urcorner, v) \wedge \psi(v)),$$

que es una sentencia de $\mathcal{L}$, y es aritmética si lo es ψ. Observemos que ϕ resulta de sustituir en δ su variable libre u por $\ulcorner\delta\urcorner \equiv 0^{(\delta)}$, luego se cumple $\sigma_0(\delta, \phi)$, luego, según el teorema 6.29,

$$\underset{Q}{\vdash} \bigwedge v(\sigma(\ulcorner\delta\urcorner, v) \leftrightarrow v = \ulcorner\phi\urcorner)$$

y, como T representa a Q, tenemos también que

$$\underset{T}{\vdash} \bigwedge v \in \mathbb{N}(\bar{\sigma}(\ulcorner\delta\urcorner, v) \leftrightarrow v = \ulcorner\phi\urcorner),$$

donde ahora $\ulcorner\delta\urcorner$ y $\ulcorner\phi\urcorner$ representan numerales de $\mathcal{L}$ y no de $\mathcal{L}_a$. En particular,

$$\underset{T}{\vdash} \bar{\sigma}(\ulcorner\delta\urcorner, \ulcorner\phi\urcorner).$$

Pero, uniendo estos hechos con la definición de ϕ, es inmediato que

$$\underset{T}{\vdash} (\phi \leftrightarrow \psi(\ulcorner\phi\urcorner)). \qquad \blacksquare$$

Nota Observemos que la prueba se adapta fácilmente al caso en que la fórmula dada tenga un parámetro $\psi(x, p)$, en cuyo caso obtenemos una fórmula $\phi(p)$ tal que $\underset{T}{\vdash} \bigwedge p(\phi(p) \leftrightarrow \psi(\ulcorner\phi\urcorner, p))$. En efecto, en estas condiciones tenemos una fórmula

$$\delta(u, p) \equiv \bigvee y \in \mathbb{N}(\bar{\sigma}(u, y) \wedge \psi(y, p)) \in \mathrm{Form}(\mathcal{L}),$$

a partir de la cual construimos

$$\phi(p) \equiv \delta(\ulcorner\delta\urcorner, p) \equiv \bigvee y \in \mathbb{N}(\bar{\sigma}(\ulcorner\delta\urcorner, y) \wedge \psi(y, p)),$$

y el razonamiento vale sin cambio alguno. $\blacksquare$

Observaciones La fórmula $\bar{\sigma}(u,v)$ que hemos construido significa "v es la fórmula que resulta de sustituir en la fórmula u la variable $\ulcorner u\urcorner$ por el numeral $0^{(u)}$", de modo que $\delta(u)$ significa "la fórmula v que resulta de sustituir en la fórmula u la variable $\ulcorner u\urcorner$ por el numeral $0^{(u)}$ tiene la propiedad $\psi(v)$" y, por consiguiente, ϕ significa "la fórmula que resulta de sustituir en la fórmula $\ulcorner\delta\urcorner$ la variable $\ulcorner u\urcorner$ por el numeral $0^{(\ulcorner\delta\urcorner)}$ tiene la propiedad ψ", pero sucede que la fórmula que cumple dicha descripción es precisamente $\ulcorner\phi\urcorner$, luego así hemos conseguido una fórmula ϕ que hable de sí misma (o, más precisamente, del numeral $\ulcorner\phi\urcorner$) y diga cualquier cosa prefijada que queramos que diga.

También es conveniente observar que la prueba del teorema anterior es totalmente constructiva. De hecho, esto está garantizado *a priori* por el hecho de que la hemos formalizado en $\mathrm{I}\Sigma_1$ (y, dado que el enunciado del teorema —para una teoría T prefijada— es Σ_1, de hecho se puede probar en ARP), pero un análisis de la construcción de ϕ dada en la prueba muestra que, en efecto, sabemos cómo definirla explícitamente y, si sabemos demostrar en T las traducciones de los axiomas de Q, también podemos obtener una demostración explícita en T de la equivalencia del enunciado. ∎

El interés del teorema anterior se pone de manifiesto cuando lo usamos para construir sentencias que afirmen de sí mismas hechos "comprometidos":

Definición 9.2 Si T es una teoría semirrecursiva que interprete a Q, podemos considerar la fórmula $\psi(\alpha) \equiv \neg \underset{\ulcorner T\urcorner}{\vdash} \alpha$ de $\mathcal{L}_a$, así como su traducción a $\mathcal{L}$, que representaremos con la misma notación. Por el teorema anterior, existe una sentencia aritmética G del lenguaje $\mathcal{L}$ de T tal que

$$\underset{T}{\vdash}(G \leftrightarrow \neg \underset{\ulcorner T\urcorner}{\vdash} \ulcorner G\urcorner).$$

A cualquier sentencia de $\mathcal{L}_a$ que cumpla esta propiedad se le llama *sentencia de Gödel* para la teoría T. Se trata de una sentencia que afirma de sí misma que no es demostrable en T. Notemos que G es Π_1 en T, porque la fórmula derecha de la equivalencia anterior es trivialmente Π_1.

Aunque no es exactamente lo mismo, el hecho de que G afirme su propia indemostrabilidad implica que ciertamente es indemostrable:

Teorema 9.3 (Primer teorema de incompletitud de Gödel) *Sea T una teoría semirrecursiva consistente que interprete a* Q*. Entonces la sentencia de Gödel de T no es demostrable en T.*

Demostración: Si $\underset{T}{\vdash} G$, entonces $\underset{\mathrm{Q}}{\vdash} \underset{\ulcorner T\urcorner}{\vdash} \ulcorner G\urcorner$ (teorema 6.32), luego $\underset{T}{\vdash} \underset{\ulcorner T\urcorner}{\vdash} \ulcorner G\urcorner$, donde la última fórmula es la traducción a $\mathcal{L}$ de la anterior, luego, por definición de sentencia de Gödel, $\underset{T}{\vdash} \neg G$, y resulta que T es contradictoria. ∎

Gödel demostró el teorema que hoy lleva su nombre añadiendo una hipótesis sobre T que no sólo garantiza que G no es demostrable, sino también que no es refutable, con lo que la teoría T resulta ser incompleta. A dicha hipótesis la llamó ω-consistencia, pero enseguida vamos a ver que no es necesaria en realidad.

No obstante, observemos que hay una hipótesis sencilla que garantiza que G no es ni demostrable ni refutable en T, a saber, que la teoría T no permita demostrar (traducciones a $\mathcal{L}$ de) fórmulas de tipo Σ_1 que sean falsas (lo cual ya implica que T es consistente).

En efecto, el teorema anterior afirma que $\neg \underset{T}{\vdash} G$, luego, según el teorema 6.15, se cumple $\mathbb{N} \vDash_{\Pi_1} \neg \underset{\ulcorner T \urcorner}{\vdash} \ulcorner G \urcorner$, luego

$$\neg \mathbb{N} \vDash_{\Sigma_1} \underset{\ulcorner T \urcorner}{\vdash} \ulcorner G \urcorner.$$

Si $\neg G$ fuera demostrable en T, tendríamos

$$\underset{T}{\vdash} \underset{\ulcorner T \urcorner}{\vdash} \ulcorner G \urcorner,$$

con lo que tendríamos una sentencia falsa de tipo Σ_1 demostrable en T.

Con esto obtenemos lo que ya sabíamos por el teorema 8.41: en toda teoría semirrecursiva consistente que interprete a Q hay una sentencia aritmética G (de tipo Π_1) que no es demostrable y además es verdadera en la interpretación natural, por lo que, si la teoría no permite demostrar sentencias falsas (de tipo Σ_1), la sentencia G tampoco es refutable.

La prueba es completamente constructiva: podemos programar a un ordenador para que, si le damos una demostración de G en T, éste nos devuelva una demostración de $\neg G$ en T que pruebe que T es contradictoria.

En particular vemos que el teorema de Σ_1-completitud de Q (teorema 6.17) no puede generalizarse a fórmulas de tipo Π_1 ni para Q ni para ninguna teoría que interprete (en particular que extienda) a Q.

Veamos ahora que en realidad no hace falta ninguna hipótesis adicional para garantizar la incompletitud de una teoría en las condiciones del teorema anterior:

Teorema 9.4 (Teorema de incompletitud (versión de Rosser)) *Toda teoría semirrecursiva consistente que interprete a* Q *es incompleta.*

Demostración: Sea T una teoría en las hipótesis del enunciado sobre un lenguaje formal $\mathcal{L}$. Entonces la fórmula $\underset{T}{\vdash} \alpha$ es Σ_1 (en el metalenguaje $\mathcal{L}_a^0$), luego es equivalente a una fórmula[1] $\bigvee d\, \sigma(d, \alpha)$, donde σ es una fórmula de tipo Δ_0. Podemos considerar la fórmula $\ulcorner \sigma \urcorner$ de $\mathcal{L}_a$ y su traducción $\bar{\sigma}$ a $\mathcal{L}$. Sea R una sentencia dada por el teorema 9.1 de modo que[2]

$$\underset{T}{\vdash} (R \leftrightarrow \bigwedge d \in \mathbb{N}\ (\bar{\sigma}(d, \ulcorner R \urcorner) \to \bigvee e \leq d\ \bar{\sigma}(e, \neg \ulcorner R \urcorner))).$$

---

[1] Si la teoría T es recursiva, podemos tomar $\sigma(d, \alpha) \equiv \overset{d}{\underset{\ulcorner T \urcorner}{\vdash}} \alpha$, que no es Δ_0, sino Δ_1, pero si la tomamos en las condiciones del teorema 6.28, toda la demostración vale sin cambio alguno, y entonces d tiene una interpretación directa: es (un número natural que codifica) una demostración de α.

[2] Notemos que R afirma de sí misma algo así como "Si puedo ser demostrada, también puedo ser refutada (con una refutación menor que mi demostración)", por lo que indirectamente R afirma que no es demostrable supuesto que la teoría T sea consistente. Notemos también que, como la sentencia de Gödel, R es de tipo Π_1.

Notemos que R es aritmética, es decir, es la traducción a $\mathcal{L}$ de una sentencia de $\mathcal{L}_a$. Vamos a probar que R no es demostrable ni refutable en T. Supongamos que se cumple $\underset{T}{\vdash} R$. Entonces existe un d tal que $\sigma(d, R)$, luego, por 6.18, se cumple $\underset{Q}{\vdash} \ulcorner\sigma\urcorner(0^{(d)}, \ulcorner R\urcorner)$ y, como T interpreta a Q, $\underset{T}{\vdash} \bar{\sigma}(0^{(d)}, \ulcorner R\urcorner)$. Por la construcción de R, esto implica que $\underset{T}{\vdash} \bigvee e \leq 0^{(d)}\ \sigma(e, \neg\ulcorner R\urcorner)$.

Consideramos ahora todos los números $e \leq d$. Si existe uno que cumpla $\sigma(e, \neg R)$, entonces $\underset{T}{\vdash} \neg R$, y tenemos que T es contradictoria. En caso contrario, para todo $e \leq d$ se cumple $\neg\sigma(e, \neg R)$, luego $\underset{Q}{\vdash} \neg\ulcorner\sigma\urcorner(0^{(e)}, \neg\ulcorner R\urcorner)$ y, usando 6.4, concluimos que $\underset{Q}{\vdash} \bigwedge e \leq 0^{(d)}\ \neg\ulcorner\sigma\urcorner(e, \neg\ulcorner R\urcorner)$, luego $\underset{T}{\vdash} \bigwedge e \leq 0^{(d)}\ \neg\bar{\sigma}(e, \neg\ulcorner R\urcorner)$, y tenemos de nuevo una contradicción en T.

Supongamos ahora $\underset{T}{\vdash} \neg R$. Entonces existe un e tal que $\sigma(e, \neg R)$, de donde $\underset{Q}{\vdash} \ulcorner\sigma\urcorner(0^{(e)}, \neg\ulcorner R\urcorner)$ y a su vez $\underset{T}{\vdash} \bar{\sigma}(0^{(e)}, \neg\ulcorner R\urcorner)$. Por otro lado, por la construcción de R tenemos que

$$\underset{T}{\vdash} \bigvee d \in \mathbb{N}\ (\sigma(d, \ulcorner R\urcorner) \wedge \bigwedge e \leq d\ \neg\sigma(e, \neg\ulcorner R\urcorner)).$$

Razonando en T, tomamos un d que cumpla esto. En Q (luego en T) se demuestra $0^{(e)} \leq d \vee d \leq 0^{(e)}$, pero el primer caso lleva a una contradicción, luego tiene que ser $d \leq 0^{(e)}$. Nuevamente tenemos dos posibilidades: si algún número $i \leq e$ cumple $\sigma(i, R)$, entonces $\underset{T}{\vdash} R$ y T resulta ser contradictoria. En caso contrario, para todo $i \leq e$ se cumple $\neg\sigma(i, R)$, luego $\underset{Q}{\vdash} \neg\ulcorner\sigma\urcorner(0^{(i)}, \ulcorner R\urcorner)$ y, usando 6.4, concluimos $\underset{Q}{\vdash} \bigwedge i \leq 0^{(e)} \neg\ulcorner\sigma\urcorner(i, \ulcorner R\urcorner)$, de donde $\underset{T}{\vdash} \bigwedge i \leq 0^{(e)} \neg\bar{\sigma}(i, \ulcorner R\urcorner)$, luego en particular $\neg\bar{\sigma}(d, \ulcorner R\urcorner)$, y tenemos de nuevo una contradicción. ∎

El lector debería convencerse de que la prueba del teorema anterior también es constructiva: dada una teoría semirrecursiva que interprete a Q, podemos construir explícitamente la sentencia R, y podemos programar a un ordenador para que si le damos una prueba de R o de $\neg R$ en T, nos devuelva la prueba de una contradicción en T.

De este modo, si una teoría axiomática cumple los requisitos mínimos de ser semirrecursiva y consistente (sin lo cual sería inútil en la práctica) y de demostrar unas mínimas propiedades sobre los números naturales (los axiomas de Q), entonces es incompleta: ninguna teoría axiomática aceptable para un matemático puede resolver cualquier problema aritmético, en el sentido de que siempre habrá sentencias aritméticas que no podrán ser demostradas ni refutadas.

Más aún: el teorema siguiente muestra que no sólo no podemos demostrar o refutar cualquier sentencia, sino que no siempre podemos saber si una sentencia dada es demostrable o no en una teoría dada.

Teorema 9.5 *El conjunto de los teoremas de una teoría consistente que interprete a* Q *no es recursivo.*

DEMOSTRACIÓN: En principio, este teorema tiene que enunciarse y demostrarse en ACR_0 (que es donde hemos desarrollado la teoría básica de la recursión), pero el teorema 8.18 permite probar en ACR_0 que todo conjunto recursivo es Δ_1, y si T es una teoría en las condiciones del enunciado, el hecho de que el conjunto de los teoremas de T no es Δ_1 puede probarse, de hecho, en $\mathrm{I}\Sigma_1$.

En efecto, sea $\phi(x)$ una fórmula Σ_1 en las condiciones del teorema 6.28 que defina al conjunto de los teoremas de T, es decir (usando que T interpreta a Q), que cumple

$$\text{si } \underset{T}{\vdash} \alpha \quad \text{entonces} \quad \underset{T}{\vdash} \phi(\ulcorner\alpha\urcorner) \quad \text{y si no } \underset{T}{\vdash} \alpha \quad \text{entonces} \quad \underset{T}{\vdash} \neg\phi(\ulcorner\alpha\urcorner).$$

El teorema 9.1 nos da una sentencia C tal que

$$\underset{T}{\vdash} (C \leftrightarrow \neg\phi(\ulcorner C\urcorner)).$$

Nuevamente, C es una sentencia que afirma que no puede ser demostrada. Si esto fuera cierto, es decir, si no $\underset{T}{\vdash} C$, entonces $\underset{T}{\vdash} \neg\phi(\ulcorner C\urcorner)$, pero por construcción de C esto equivale a $\underset{T}{\vdash} C$, y tenemos una contradicción. Por lo tanto, $\underset{T}{\vdash} C$, luego $\underset{T}{\vdash} \phi(\ulcorner C\urcorner)$, pero esto implica $\underset{T}{\vdash} \neg C$, y llegamos a que T es contradictoria. ∎

Tenemos así un ejemplo conceptualmente muy simple de conjunto semirrecursivo no recursivo: el conjunto de los teoremas de cualquier teoría semirrecursiva consistente que interprete a Q (por ejemplo el conjunto de los teoremas de la propia Q, o de AP). En general disponemos de un algoritmo finito para enumerar todos los teoremas de una teoría axiomática recursiva T (sólo tenemos que ir enumerando los números naturales, comprobando si cada uno de ellos codifica una demostración de la teoría y, en caso afirmativo, añadir a la lista de teoremas la conclusión de tal demostración). Si queremos confirmar que una determinada fórmula es un teorema de T, en teoría siempre tenemos el "método" (muy poco práctico) de esperar a ver si aparece en la lista, pero ahora acabamos de probar que, si una fórmula no es un teorema de una teoría semirrecursiva consistente T, no existe ningún algoritmo que nos permita confirmar en un tiempo finito que no es un teorema (lo cual no impide que en casos concretos logremos averiguarlo por uno u otro medio, pero no existe un procedimiento general que nos garantice una respuesta en cualquier caso).

Ejercicio: Modificar la prueba del teorema anterior para probar que el conjunto de sentencias de tipo Π_1 demostrables en T no es recursivo.

Un argumento que nos convenza de que una afirmación es cierta no tiene por qué ser demostrable a partir de unos axiomas dados. Sin embargo, el teorema siguiente (demostrable en ACR_0) muestra que la imposibilidad de saber si una afirmación es cierta o falsa no depende de los axiomas que fijemos:

Teorema 9.6 *Si M es un modelo de una teoría axiomática T sobre un lenguaje formal $\mathcal{L}$ que interprete a* Q*, el conjunto de las sentencias α de tipo Σ_1 (o Π_1) de $\mathcal{L}_a$ cuya traducción $\bar{\alpha}$ a $\mathcal{L}$ cumple $M \models \bar{\alpha}$ no es recursivo.*

DEMOSTRACIÓN: Supongamos que el conjunto Γ indicado en el enunciado (para sentencias Π_1) fuera recursivo. Entonces podemos considerar la teoría axiomática T' sobre $\mathcal{L}_a$ cuyos axiomas son los de Q (considerados como sentencias, cuantificando universalmente sus variables libres) más las sentencias $\alpha \in \Gamma$ más las sentencias $\neg\alpha$, donde α es de tipo Π_1, pero $\alpha \notin \Gamma$.

Es claro que T' es una teoría axiomática recursiva que interpreta a Q, y es consistente, pues las traducciones de todos los axiomas de T' son verdaderas en M, luego lo mismo sucede con las traducciones de sus teoremas, luego $0 \neq 0$ no puede ser un teorema de T'. Por el teorema 9.4 debería haber una sentencia R (que en la prueba se ve que es de tipo Π_1) que no fuera ni demostrable ni refutable en T', pero si $M \vDash \bar{R}$ entonces $R \in \Gamma$ es un axioma de T', luego es un teorema, y si $\neg M \vDash \bar{R}$, entonces $\neg R$ es un axioma de T', luego R es refutable en T', y así tenemos una contradicción.

Esto prueba el teorema para sentencias Π_1, pero es claro que de aquí se sigue el caso para sentencias Σ_1, pues si Γ' es el conjunto correspondiente a sentencias Σ_1, es claro que $\alpha \equiv \bigwedge u\delta \in \Gamma$ si y sólo si $\bigvee u\neg\delta \in \Gamma'$, de donde se sigue fácilmente que si Γ' fuera recursivo también lo sería Γ. ∎

El teorema 9.5 todavía puede generalizarse más hasta eliminar toda alusión explícita a los números naturales:

Teorema 9.7 (Teorema de Church) *El conjunto de los teoremas lógicos de un lenguaje formal que contenga al menos un relator diádico distinto del igualador no es recursivo.*

DEMOSTRACIÓN: Como en el caso del teorema anterior, usando que en ACR_0 se puede probar que todo conjunto recursivo es Δ_1, sólo tenemos que probar que el conjunto de los teoremas lógicos de un lenguaje formal en las condiciones del enunciado no es Δ_1, y esto puede probarse en $\mathrm{I}\Sigma_1$.

Consideremos el lenguaje $\mathcal{L}_{\mathrm{tc}}$ de la teoría de conjuntos, y consideramos cualquier teoría de conjuntos consistente que interprete a Q, como por ejemplo, la teoría de Kripke-Platek descrita en la sección 5.4, cuya consistencia equivale a la de $\mathrm{I}\Sigma_1$, luego a la de ARP.

Puesto que Q consta de un número finito de axiomas propios, podemos considerar la teoría T que consta de un único axioma C formado por la conjunción de los axiomas de KP necesarios para demostrar las traducciones a $\mathcal{L}_{\mathrm{tc}}$ de los axiomas de Q, y así T es una teoría sobre $\mathcal{L}_{\mathrm{tc}}$ con un único axioma que interpreta a Q. Más en general, podemos considerar a T como teoría axiomática sobre cualquier lenguaje formal $\mathcal{L}$ que tenga un relator diádico distinto del igualador, identificándolo con el relator $\in$ de $\mathcal{L}_{\mathrm{tc}}$, de modo que los axiomas de T no incluyen ningún otro signo eventual que pueda tener $\mathcal{L}$.

Podemos suponer que C es una sentencia (cuantificando universalmente sus variables libres, si las hay). Entonces

$$\underset{T}{\vdash} \alpha \quad \leftrightarrow \quad \vdash C \to \alpha.$$

Si la fórmula $\vdash \alpha$ fuera de tipo Δ_1, es claro que también lo sería la fórmula $\phi(\alpha) \equiv \vdash (C \to \alpha)$, es decir, que el conjunto de los teoremas de T sería recursivo, en contradicción con el teorema anterior. ∎

Notemos que el teorema vale igualmente para lenguajes formales con un relator n-ádico, con $n \geq 2$ (distinto del igualador), pues éste permite formular igualmente los axiomas de KP haciendo que los argumentos posteriores a los dos primeros sean irrelevantes. En cambio:

Ejercicio: Probar que si una fórmula contiene únicamente relatores monádicos (aparte del igualador) entonces existe un algoritmo finito para determinar si es o no consistente o si es o no un teorema lógico. AYUDA: Probar que si una fórmula con n relatores monádicos tiene un modelo, entonces tiene un modelo con a lo sumo 2^n elementos. Para ello basta establecer una relación de equivalencia en el modelo dado en la que dos objetos de su universo son equivalentes si las n relaciones que interpretan los relatores coinciden en ellos. Las clases de equivalencia se convierten fácilmente en un modelo de la fórmula dada, y son a lo sumo 2^n. Nótese además que sólo hay un número finito de modelos "esencialmente distintos" con a lo sumo 2^n elementos y todos ellos pueden ser calculados en la práctica.

Así pues, no existe ningún criterio que permita distinguir en un tiempo finito si cualquier fórmula dada de un lenguaje formal es un teorema lógico, es consistente o es contradictoria (porque una fórmula es consistente si y sólo si su negación no es un teorema lógico, y es contradictoria si y sólo si su negación es un teorema lógico).

9.2 El segundo teorema de incompletitud

Nos ocupamos ahora de demostrar el segundo teorema de incompletitud de Gödel, cuya prueba consiste esencialmente de formalizar la demostración del primer teorema de incompletitud. Supongamos que T es una teoría semirrecursiva que interprete a Q, sea G una sentencia de Gödel para T y consideremos la sentencia de tipo Π_1

$$\tilde{G} \equiv \neg \underset{\ulcorner T \urcorner}{\vdash} \ulcorner G \urcorner.$$

Podemos considerar a $\tilde{G}$ como fórmula de $\mathcal{L}_a$ (traducible al lenguaje $\mathcal{L}$ de T), de modo que el hecho de que G sea una sentencia de Gödel significa que

$$\underset{T}{\vdash}(G \leftrightarrow \tilde{G})$$

(aunque, tal y como hemos construido G, esta equivalencia se cumple, de hecho, en $\mathrm{I}\Sigma_1$). Por otro lado, tras el teorema 6.21 hemos definido la sentencia de tipo Π_1:

$$\operatorname{Consis} T \equiv \neg \underset{T}{\vdash} \ulcorner 0 \neq 0 \urcorner,$$

que expresa la consistencia de T. En estos términos, el teorema 9.3 afirma que

$$\operatorname{Consis} T \to \neg \underset{T}{\vdash} G.$$

mientras que la implicación contraria es trivial, pues $\tilde{G}$ afirma que una sentencia no es demostrable en T, y eso implica la consistencia de T. Así pues, tenemos que

$$\operatorname{Consis} T \leftrightarrow \neg \underset{T}{\vdash} G.$$

Esto lo hemos demostrado en $\mathrm{I}\Sigma_1$. La clave del segundo teorema de incompletitud es que en $\mathrm{I}\Sigma_1$ podemos demostrar que esto es demostrable en $\mathrm{I}\Sigma_1$, donde el segundo $\mathrm{I}\Sigma_1$ es el definido formalmente en el primero. Con la notación que venimos empleando, el primero sería la metateoría sobre $\mathcal{L}_a^0$ y el segundo el definido sobre $\mathcal{L}_a$.

Pero la formalización de esta equivalencia es:

$$\underset{\mathrm{I}\Sigma_1}{\vdash} \mathrm{Consis}\,\ulcorner T\urcorner \leftrightarrow \tilde{G},$$

que a su vez equivale a

$$\underset{\mathrm{I}\Sigma_1}{\vdash} \mathrm{Consis}\,\ulcorner T\urcorner \leftrightarrow G,$$

y esto significa que la sentencia de Gödel de una teoría T equivale a la consistencia de la formalización $\ulcorner T\urcorner$ de T en $\mathrm{I}\Sigma_1$. Si suponemos que T interpreta a $\mathrm{I}\Sigma_1$, de aquí podemos pasar a

$$\underset{T}{\vdash} \mathrm{Consis}\,\ulcorner T\urcorner \leftrightarrow G,$$

pero el teorema de incompletitud afirma que, si T es consistente, entonces G no es demostrable en T, luego $\mathrm{Consis}\,T$ tampoco, y esto es el segundo teorema de incompletitud. Con detalle:

Teorema 9.8 (Segundo teorema de incompletitud de Gödel) *Si T es una teoría semirrecursiva que interpreta a $\mathrm{I}\Sigma_1$ y G es una sentencia de Gödel para T, entonces*

$$\underset{T}{\vdash} (\mathrm{Consis}\,\ulcorner T\urcorner \leftrightarrow G).$$

En particular, si T es consistente no $\underset{T}{\vdash} \mathrm{Consis}\,\ulcorner T\urcorner$.

Demostración: En realidad vamos a probar lo que afirmábamos en los comentarios previos al teorema, que es ligeramente más fuerte: si llamamos $\tilde{G} \equiv \neg \underset{\ulcorner T\urcorner}{\vdash} \ulcorner G\urcorner$, demostraremos que

$$\underset{\mathrm{I}\Sigma_1}{\vdash} (\mathrm{Consis}\,\ulcorner T\urcorner \leftrightarrow \tilde{G}),$$

entendiendo ambos términos como sentencias de $\mathcal{L}_a$. Admitiendo esto, la traducción de esta equivalencia al lenguaje de T es demostrable en T, pero en T se cumple que $\tilde{G}$ es equivalente a G (por definición de sentencia de Gödel), luego tenemos la equivalencia del enunciado.

La prueba se basa exclusivamente en las condiciones de Hilbert-Bernays demostradas en 6.32.

Como $\underset{T}{\vdash} (0 \neq 0 \to G)$, la propiedad 1. del teorema 6.32 nos da que

$$\underset{\mathrm{I}\Sigma_1}{\vdash} \underset{\ulcorner T\urcorner}{\vdash} \ulcorner 0 \neq 0 \to G\urcorner,$$

luego la propiedad 3. nos da que

$$\underset{\mathrm{I}\Sigma_1}{\vdash} \underset{\ulcorner T\urcorner}{\vdash} \ulcorner 0 \neq 0\urcorner \to \underset{\ulcorner T\urcorner}{\vdash} \ulcorner G\urcorner,$$

luego, aplicando la regla de Negación del Implicador, concluimos que

$$\underset{\mathrm{I}\Sigma_1}{\vdash} \tilde{G} \to \mathrm{Consis}\ulcorner T\urcorner.$$

Recíprocamente, por la propiedad 2. del teorema 6.32,

$$\underset{\mathrm{I}\Sigma_1}{\vdash} (\underset{\ulcorner T\urcorner}{\vdash} \ulcorner G\urcorner \to \underset{\ulcorner T\urcorner}{\vdash} \ulcorner \underset{\ulcorner T\urcorner}{\vdash} \ulcorner G\urcorner\urcorner),$$

que es lo mismo que

$$\underset{\mathrm{I}\Sigma_1}{\vdash} (\neg\tilde{G} \to \underset{\ulcorner T\urcorner}{\vdash} \ulcorner \underset{\ulcorner T\urcorner}{\vdash} \ulcorner G\urcorner\urcorner).$$

Por definición de sentencia de Gödel, $\underset{T}{\vdash} \underset{\ulcorner T\urcorner}{\vdash} \ulcorner G\urcorner \to \neg G$, luego por 1.,

$$\underset{\mathrm{I}\Sigma_1}{\vdash} \underset{\ulcorner T\urcorner}{\vdash} \ulcorner \underset{\ulcorner T\urcorner}{\vdash} \ulcorner G\urcorner \to \neg G\urcorner$$

y por 3.,

$$\underset{\mathrm{I}\Sigma_1}{\vdash} (\neg\tilde{G} \to \underset{\ulcorner T\urcorner}{\vdash} \ulcorner \neg G\urcorner).$$

Por otro lado, por definición de $\tilde{G}$ tenemos trivialmente que

$$\underset{\mathrm{I}\Sigma_1}{\vdash} (\neg\tilde{G} \to \underset{\ulcorner T\urcorner}{\vdash} \ulcorner G\urcorner),$$

luego por la formalización de la regla de contradicción

$$\underset{\mathrm{I}\Sigma_1}{\vdash} (\neg\tilde{G} \to \underset{\ulcorner T\urcorner}{\vdash} \ulcorner 0 \neq 0\urcorner),$$

es decir, $\underset{\mathrm{I}\Sigma_1}{\vdash} (\neg\tilde{G} \to \neg\,\mathrm{Consis}\ulcorner T\urcorner)$. ∎

Nota Al exigir que la teoría T interprete a $\mathrm{I}\Sigma_1$, el teorema anterior no prueba que $\mathrm{Consis}\ulcorner Q\urcorner$ no sea demostrable en Q. Sucede que esto es cierto, pero no vamos a ver aquí la prueba.[3] ∎

Observaciones Los teoremas de incompletitud han dado pie a muchas interpretaciones erróneas que, esencialmente, parten de la creencia de que lo que prueban es que la sentencia G es verdadera, pero no demostrable, lo cual da pie a especular sobre cómo podemos saber que es verdadera si estamos diciendo que no se puede demostrar, pero esto no es así, y ello se ve más claramente si, en lugar de considerar G, consideramos la sentencia equivalente $\mathrm{Consis}\ulcorner T\urcorner$. Lo que afirman los teoremas de incompletitud no es que $\mathrm{Consis}\ulcorner T\urcorner$ es verdadera y no demostrable (en T), sino que si $\mathrm{Consis}\ulcorner T\urcorner$ es verdadera, entonces no es demostrable (en T). Ahora bien, nos podemos encontrar en dos casos distintos:

[3]Pudlák, P. *Cuts, Consistency Statements and Interpretations,* Journal of Symbolic Logic (1985) 50 (2), pp. 423–441.

1. Puede ocurrir que tengamos un argumento que nos convenza de que la teoría T es verdadera. En ese caso, el teorema de incompletitud nos asegura que ese argumento no será formalizable en la propia teoría T, sin perjuicio de que lo sea en otra teoría más potente. Así, la consistencia de $\mathrm{I}\Sigma_1$ se puede demostrar en $\mathrm{I}\Sigma_2$, o en AP, etc.

2. La alternativa es que no tengamos ningún argumento que nos convenza de la consistencia de T. Este caso se tiene que dar necesariamente cuando T es una teoría lo suficientemente potente como para formalizar cualquier argumento finitista. Por ejemplo, podemos pensar en la teoría de conjuntos de Zermelo-Fraenkel ZFC, que hemos descrito brevemente al final de la sección 5.5. Aunque no hemos profundizado en ella (por su naturaleza esencialmente no finitista), se trata de la teoría más usada como fundamentación de toda la matemática, de modo que en ella se puede formalizar cualquier argumento que un matemático considere convincente. Por lo tanto, si existiera algún argumento que justificara la consistencia de ZFC, dicho argumento tendría que poder formalizarse en ZFC, y ello contradiría el segundo teorema de incompletitud. La conclusión es que es imposible demostrar (convincentemente) la consistencia de ZFC y, más en general, de cualquier teoría axiomática capaz de formalizar cualquier argumento matemático.

   La precisión "convincentemente" es esencial, pues, por ejemplo, si a ZFC le añadimos como axioma Consis ZFC obtenemos una teoría axiomática en la que podemos demostrar obviamente Consis $\ulcorner$ZFC$\urcorner$, pero la "demostración" no nos da ninguna garantía de que ZFC sea consistente. Hay muchas teorías más fuertes que ZFC en las que es posible demostrar la consistencia de ZFC, pero una prueba formal de la consistencia de ZFC en el seno de una teoría axiomática cuya consistencia es más dudosa aún que la de ZFC no proporciona ninguna convicción de que ZFC sea consistente.[4] Si ZFC es consistente, entonces Consis ZFC es un ejemplo de afirmación verdadera que nunca podremos demostrar "convincentemente" que lo es.

Veamos una aplicación del segundo teorema de incompletitud:

Teorema 9.9 *Ninguna extensión consistente de la Aritmética de Peano es finitamente axiomatizable.*

Demostración: Observemos que un conjunto finito de axiomas puede reducirse a un único axioma formando su conjunción. Se trata de probar que si una sentencia γ de $\mathcal{L}_a$ implica todos los axiomas de AP entonces es contradictoria. Pongamos que la sentencia es Σ_n. Vamos a probar que $\gamma \vdash \mathrm{Consis}\,\ulcorner\gamma\urcorner$, lo cual, por el segundo teorema de incompletitud, implica que γ es contradictoria.

[4] No obstante, prácticamente todos los expertos en teoría de conjuntos están convencidos de que ZFC es consistente, simplemente porque no hay ninguna razón para sospechar que no lo es. El segundo teorema de incompletitud implica, precisamente, que el hecho de que no se disponga de ninguna prueba "convincente" de que lo es no es un motivo para sospechar que no vaya a serlo.

Suponemos, pues γ y, por reducción al absurdo, suponemos $\neg\,\mathrm{Consis}\ulcorner\gamma\urcorner$, es decir, suponemos que $\ulcorner\gamma\urcorner \vdash \ulcorner 0 \neq 0\urcorner$. Esto implica $\vdash \ulcorner\gamma \to 0 \neq 0\urcorner$ y, en particular, $\underset{\ulcorner \mathrm{I}\Sigma_n\urcorner}{\vdash} \ulcorner\gamma \to 0 \neq 0\urcorner$. Por el teorema 6.21, en $\mathrm{I}\Sigma_{n+1}$, luego en particular a partir de γ, podemos concluir $\gamma \to 0 \neq 0$ y, de nuevo porque estamos suponiendo γ, obtenemos que $0 \neq 0$. Esta contradicción prueba $\mathrm{Consis}\ulcorner\gamma\urcorner$. ∎

Nota Hay un argumento más directo para probar que AP no es finitamente axiomatizable: si lo fuera, todos sus axiomas (luego todos sus teoremas) serían teoremas de $\mathrm{I}\Sigma_n$, para un n suficientemente grande, pero entonces tendríamos que

$$\underset{\mathrm{I}\Sigma_n}{\vdash} \mathrm{Consis}\ulcorner \mathrm{I}\Sigma_n\urcorner,$$

y el teorema de incompletitud nos daría que $\mathrm{I}\Sigma_n$ es contradictorio. ∎

Nuestra segunda aplicación del segundo teorema de incompletitud es una curiosidad:

La interpretación natural de una sentencia de Gödel equivale a su no demostrabilidad, y hemos demostrado que, bajo las hipótesis triviales del teorema, las sentencias de Gödel son verdaderas (afirman que no pueden ser demostradas y, efectivamente, no pueden ser demostradas). Por simple curiosidad, podemos preguntarnos qué sucede si aplicamos el teorema 9.1 para construir una sentencia que afirma su propia demostrabilidad. Específicamente, dada una teoría semirrecursiva T que interprete a Q, sabemos construir una sentencia H tal que

$$\underset{T}{\vdash}(H \leftrightarrow \underset{\ulcorner T\urcorner}{\vdash} \ulcorner H\urcorner).$$

Las sentencias con esta propiedad se llaman *sentencias de Henkin* y no es evidente en principio si son verdaderas o falsas o —lo que en este caso es lo mismo—, si son demostrables o no. La respuesta es que todas son verdaderas, pero la prueba se basa en el segundo teorema de incompletitud.

Teorema 9.10 (Teorema de Löb) *Sea T una teoría semirrecursiva que interprete a $\mathrm{I}\Sigma_1$ y sea H una sentencia tal que*

$$\underset{T}{\vdash} (\underset{\ulcorner T\urcorner}{\vdash} \ulcorner H\urcorner \to H).$$

Entonces $\underset{T}{\vdash} H$.

Demostración: Sea T^* la extensión de T que resulta de añadirle el axioma $\neg H$. Si no $\underset{T}{\vdash} H$, entonces T^* es consistente. Formalizando (en $\mathrm{I}\Sigma_1$) este sencillo resultado obtenemos que $\underset{T^*}{\vdash} \neg \underset{\ulcorner T\urcorner}{\vdash} \ulcorner H\urcorner \to \mathrm{Consis}\ulcorner T^*\urcorner$.

Por el segundo teorema de incompletitud tenemos que no $\underset{T^*}{\vdash} \mathrm{Consis}\ulcorner T^*\urcorner$, luego no $\underset{T^*}{\vdash} \neg \underset{\ulcorner T\urcorner}{\vdash} \ulcorner H\urcorner$, de donde no $\underset{T}{\vdash}(\neg H \to \neg \underset{\ulcorner T\urcorner}{\vdash} \ulcorner H\urcorner)$, es decir, que no se cumple $\underset{T}{\vdash} \underset{\ulcorner T\urcorner}{\vdash} \ulcorner H\urcorner \to H$, como había que probar. ∎

9.3 El teorema de Tarski

La hipótesis de semirrecursividad en el teorema de incompletitud es claramente necesaria. Por ejemplo, podemos considerar la extensión T de la aritmética de Peano que resulta de tomar como axiomas todas las sentencias verdaderas en su modelo natural. Es inmediato que T cumple todos los requisitos del primer teorema de incompletitud salvo quizá la semirrecursividad y, de hecho, no puede ser semirrecursiva, pues es trivialmente completa.

El hecho de que el conjunto de las sentencias de $\mathcal{L}_a$ que cumplen $\mathbb{N} \models \alpha$ no es semirrecursivo lo razonamos ya en la nota posterior al teorema 8.41. La prueba utiliza dicho teorema, que es demostrable en ACA_0, pero ella misma no es demostrable en ACA_0, pues, como vamos a ver a continuación, en dicha teoría no puede formalizarse la definición de $\mathbb{N} \models \alpha$, es decir, no puede definirse el modelo natural de $\mathcal{L}_a$. El teorema siguiente es demostrable en ACR_0:

Teorema 9.11 (Teorema de Tarski) *Sea T una teoría axiomática que interprete a* Q *y sea M un modelo de T.*

1. *No existe ninguna fórmula $V(x)$ con x como única variable libre y tal que para toda sentencia ϕ se cumpla*

$$M \models \phi \quad \textit{si y sólo si} \quad M \models V(\ulcorner \phi \urcorner).$$

2. *Tampoco puede existir una fórmula $V(x)$ tal que para toda sentencia ϕ se cumpla $\underset{T}{\vdash}(\phi \leftrightarrow V(\ulcorner \phi \urcorner))$.*

Demostración: Supongamos que existe la fórmula $V(x)$ según 1. Entonces el teorema 9.1 nos da una sentencia τ tal que

$$\underset{T}{\vdash}(\tau \leftrightarrow \neg V(\ulcorner \tau \urcorner)).$$

Notemos que τ significa "yo soy falsa" (en M).

Si $M \models \tau$, entonces $M \models V(\ulcorner \tau \urcorner)$, pero, por la propiedad que define a τ tendremos también que $M \models \neg\tau$, lo cual es absurdo.

Si $M \models \neg\tau$, entonces $M \models \neg V(\ulcorner \tau \urcorner)$, luego por hipótesis $M \models \tau$, y tenemos de nuevo un imposible. Así pues, no existe tal V.

Es claro que una fórmula que cumpla 2. también cumple 1. para cualquier modelo M de T, pero, suponiendo meramente que T es consistente, podemos obtener una prueba directa puramente sintáctica (que no involucre modelos). En efecto, construimos igualmente la sentencia τ, y ahora tenemos que $\underset{T}{\vdash}(\tau \leftrightarrow \neg\tau)$, de donde se sigue que T es contradictoria. ∎

Observaciones Vemos, pues, que el conjunto de las sentencias verdaderas en un modelo M de una teoría axiomática T que interprete a Q no puede definirse mediante ninguna fórmula del lenguaje $\mathcal{L}$ de T. Podríamos decir que la "verdad" en un modelo de una teoría es indefinible en el lenguaje de la propia teoría.

En particular, el conjunto de las sentencias verdaderas no puede definirse mediante una fórmula de $\mathcal{L}_a$ (pues en tal caso, también sería definible por su traducción a $\mathcal{L}$), lo cual significa que no sólo no es recursivo, sino que no es aritmético, no es definible por ninguna fórmula aritmética sin variables de segundo orden.

En particular, razonando en AR_0 podemos aplicar esto al modelo natural $\mathbb{N}$ de $\mathcal{L}_a$ (tomando como T la aritmética de Pano AP o $\mathrm{I}\Sigma_1$): el conjunto de sentencias de $\mathcal{L}_a$ verdaderas en el modelo natural no es aritmético. No puede definirse mediante ninguna fórmula de $\mathcal{L}_a$, con lo que *a fortiori* no es recursivo.

El segundo apartado del teorema de Tarski afirma que, en una teoría consistente T sobre un lenguaje $\mathcal{L}$, no es posible asignar un "significado" a cada sentencia $\alpha \in \mathrm{Form}(\ulcorner\mathcal{L}\urcorner)$ de modo que, para cada sentencia α, el significado asociado a $\ulcorner\alpha\urcorner$ sea precisamente α. No obstante, el teorema 6.16 muestra que, en el caso de $\mathcal{L}_a$, este impedimento teórico puede burlarse parcialmente restringiendo la definición a determinadas clases de sentencias (Σ_n, o Π_n). ∎

9.4 Incompletitud y aritmética no estándar

Los teoremas de incompletitud nos permiten entender mejor los modelos no estándar de la aritmética. En efecto, sea T una teoría recursiva y consistente que interprete a Q. Llamemos $S(x) \equiv \underset{T}{\overset{x}{\vdash}} \ulcorner 0 \neq 0\urcorner$, de modo que $S(x)$ significa "x es una demostración de $0 \neq 0$ en T". Por definición:

$$\mathrm{Consis}\, T \equiv \neg\bigvee x\, S(x).$$

Que T sea recursiva significa que la fórmula $S(x)$ es Δ_1, luego, por 6.28, cambiándola si es preciso por otra equivalente, podemos suponer que

$$\bigwedge x\,(S(x) \to \underset{Q}{\vdash} \ulcorner S\urcorner(0^{(x)})), \qquad \bigwedge x\,(\neg S(x) \to \underset{Q}{\vdash} \neg\ulcorner S\urcorner(0^{(x)})).$$

Que T sea consistente significa que se cumple $\mathrm{Consis}\, T$, por lo que

$$\bigwedge x\ \underset{Q}{\vdash} \neg\ulcorner S\urcorner(0^{(x)}),$$

y, como T interpreta a Q, también

$$\bigwedge x \underset{T}{\vdash} \neg\bar{S}(0^{(x)}),$$

donde $\bar{S}$ es la traducción al lenguaje $\mathcal{L}$ de T de la formalización $\ulcorner S\urcorner$ de S en $\mathcal{L}_a$. Por otro lado, podemos considerar la fórmula de $\mathcal{L}$

$$\mathrm{Consis}\,\ulcorner T\urcorner \equiv \neg\bigvee x \in \mathbb{N}\ \bar{S}(x).$$

El segundo teorema de incompletitud asegura que

$$\neg \underset{T}{\vdash} \mathrm{Consis}\,\ulcorner T\urcorner,$$

de modo que en T podemos demostrar $\neg\bar{S}(0), \neg\bar{S}(1), \neg\bar{S}(2), \ldots$, pero no podemos demostrar $\bigwedge x \in \mathbb{N}\, \neg\bar{S}(x)$. Más aún, esto hace que la teoría T' que resulta de añadir $\neg\,\mathrm{Consis}\ulcorner T\urcorner$ a los axiomas de T es consistente, por lo que

$$\bigwedge x \underset{T'}{\vdash} \neg\bar{S}(0^{(x)}), \qquad \underset{T'}{\vdash} \bigvee x \in \mathbb{N}\ \bar{S}(x).$$

Más explícitamente, en T' podemos demostrar que existe un número natural que cumple $\bar{S}(x)$, pero también podemos probar $\neg\bar{S}(0), \neg\bar{S}(1), \neg\bar{S}(2), \ldots$

Si suponemos que T interpreta a $\mathrm{I}\Sigma_1$, en T' podemos demostrar

$$\overset{1}{\bigvee} x \in \mathbb{N}(\bar{S}(x) \wedge \bigwedge y < x\, \neg\bar{S}(y)),$$

es decir, que existe un mínimo número natural que cumple $\bar{S}(x)$. Esto nos permite definir

$$c \equiv x \mid (x \in \mathbb{N} \wedge \bar{S}(x) \wedge \bigwedge y < x\, \neg\bar{S}(y)).$$

Así, c se interpreta como "la menor demostración de $0 \neq 0$ en T". Como estamos suponiendo que T es consistente, no existe tal demostración, pero vemos que es consistente suponer que exista. En estos términos, en T' podemos probar que $c \in \mathbb{N}$, pero también que $c \neq 0, c \neq 1, c \neq 2, \ldots$

Si M es un modelo de T', entonces $M(c)$ es un número natural no estándar, en el sentido que hemos introducido en la sección 7.5. Allí, para construir modelos no estándar de una teoría T tuvimos que añadir a su lenguaje una constante c sin definición, parcialmente determinada por un conjunto de axiomas adicionales que forzaban a que tuviera que interpretarse como un número no estándar. Aquí en cambio hemos definido un número natural no estándar concreto (postulando previamente su existencia al tomar como axioma $\neg\,\mathrm{Consis}\ulcorner T\urcorner$).

Observemos que c no es el mínimo número no estándar (de hecho no existe tal mínimo). En efecto, puesto que podemos probar que $c \neq 0$, de aquí deducimos que existe un número d tal que $c = d'$. Es claro que d también es no estándar, en el sentido de que para todo número natural n sabemos probar que $d \neq 0^{(n)}$.

Notemos que no existe un único modelo M para la aritmética no estándar, sino que existen infinitos modelos "no isomorfos" dos a dos, en el sentido de que satisfacen sentencias diferentes. Por ejemplo, a partir de AP podemos formar dos teorías axiomáticas consistentes, pero mutuamente contradictorias,

$$\mathrm{AP} + \mathrm{Consis}\ulcorner\mathrm{AP}\urcorner \qquad \text{y} \qquad \mathrm{AP} + \neg\,\mathrm{Consis}\ulcorner\mathrm{AP}\urcorner.$$

Si las llamamos T_0 y T_1, respectivamente, cualquiera de ellas puede extenderse a su vez a dos teorías consistentes y mutuamente contradictorias:

$$T_0 + \mathrm{Consis}\ulcorner T_0\urcorner, \quad T_0 + \neg\,\mathrm{Consis}\ulcorner T_0\urcorner, \quad T_1 + \mathrm{Consis}\ulcorner T_1\urcorner, \quad T_1 + \neg\,\mathrm{Consis}\ulcorner T_1\urcorner,$$

que podemos llamar T_{00}, T_{01}, T_{10} y T_{11}, respectivamente, e igualmente podemos formar otras ocho teorías T_{000}, T_{001}, T_{010}, ... Cada una de las teorías construidas de este modo tiene su propio modelo, y dos cualesquiera de estos modelos satisfacen sentencias mutuamente contradictorias.

Vamos a probar ahora que, en realidad, no necesitamos elegir una nueva sentencia a cada paso para formar nuevas teorías consistentes, sino que podemos usar siempre la misma. Necesitamos un resultado previo:

Teorema 9.12 *Si T es una teoría semirrecursiva consistente que interpreta a* Q*, existe una fórmula $\phi(x)$ de tipo Σ_1 tal que, para todo número natural k, la teoría*

$$T + \bigwedge x \in \mathbb{N}(\phi(x) \leftrightarrow x = 0^{(k)})$$

es consistente.

Demostración: Sea $\mathcal{L}$ el lenguaje formal de T. Consideremos la fórmula de $\mathcal{L}_a^0$ (de tipo Σ_1)

$$\chi(p,k) \equiv p \in \mathrm{Form}(\mathcal{L}) \wedge \bigvee x d(\mathrm{Vlib}(p) = \{x\} \wedge \overset{d}{\underset{T}{\vdash}} \neg\bigwedge x \in \mathbb{N}(p(x) \leftrightarrow x = 0^{(k)})$$
$$\wedge \bigwedge d' < d \bigwedge k' < d' \ \neg \overset{d'}{\underset{T}{\vdash}} \neg\bigwedge x \in \mathbb{N}(p(x) \leftrightarrow x = 0^{(k')})).$$

Así, si se cumple $\chi(p,k) \wedge \chi(p,k')$, con $k \neq k'$, existen d y d' que demuestran, respectivamente,

$$\underset{T}{\vdash} \neg\bigwedge x \in \mathbb{N}(p(x) \leftrightarrow x = 0^{(k)}), \qquad \underset{T}{\vdash} \neg\bigwedge x \in \mathbb{N}(p(x) \leftrightarrow x = 0^{(k')}).$$

No perdemos generalidad si suponemos $d' < d$, y claramente $k' < d'$ (pues d' demuestra una fórmula que contiene el numeral $0^{(k')}$), pero esto contradice la minimalidad de d que postula χ.

Esto nos permite aplicar el teorema 6.29, según el cual, cambiando χ por una fórmula equivalente, podemos suponer que si se cumple $\chi(p,k)$, entonces

$$\underset{T}{\vdash} \bigwedge v \in \mathbb{N}(\ulcorner\chi\urcorner(\ulcorner p\urcorner, v) \leftrightarrow v = 0^{(k)}).$$

Por el teorema 9.1 (véase la nota posterior), existe una fórmula $\phi(x)$ de $\mathcal{L}$ tal que

$$\underset{T}{\vdash} \bigwedge x \in \mathbb{N}(\phi(x) \leftrightarrow \ulcorner\chi\urcorner(\ulcorner\phi\urcorner, x)).$$

Esta equivalencia prueba que ϕ es Σ_1.

Basta probar que, para todo número natural k,

$$\neg \underset{T}{\vdash} \neg\bigwedge x \in \mathbb{N}(\phi(x) \leftrightarrow x = 0^{(k)}).$$

En caso contrario, consideramos el mínimo k que cumple

$$\underset{T}{\vdash} \neg\bigwedge x \in \mathbb{N}(\phi(x) \leftrightarrow x = 0^{(k)}),$$

y a su vez una demostración mínima d tal que

$$\phi \in \mathrm{Form}(\mathcal{L}) \wedge \mathrm{Vlib}(\phi) = \{x\} \wedge \overset{d}{\underset{T}{\vdash}} \neg\bigwedge x \in \mathbb{N}(\phi(x) \leftrightarrow x = 0^{(k)})$$
$$\wedge \bigwedge d' < d \ \neg \overset{d'}{\underset{T}{\vdash}} \neg\bigwedge x \in \mathbb{N}(\phi(x) \leftrightarrow x = 0^{(k)}).$$

Ahora, si $k' < d' < d$ y $\underset{T}{\overset{d'}{\vdash}} \neg\bigwedge x \in \mathbb{N}(\phi(x) \leftrightarrow x = 0^{(k')})$, tiene que ser $k' = k$, por la minimalidad de k, y esto contradice la minimalidad de d. Por lo tanto, se cumple $\chi(\phi, k)$. El teorema 6.29 nos da que

$$\underset{T}{\vdash} \bigwedge x \in \mathbb{N}(\ulcorner\chi\urcorner(\ulcorner\phi\urcorner, x) \leftrightarrow x = 0^{(k)})$$

y, por la construcción de ϕ,

$$\underset{T}{\vdash} \bigwedge x \in \mathbb{N}(\phi(x) \leftrightarrow x = 0^{(k)}),$$

y resulta que T es contradictoria. ∎

Notemos que, si ϕ es una fórmula en las condiciones del teorema anterior, para todo número natural se cumple $\neg \underset{T}{\vdash} \phi(0^{(k)})$, pues en caso contrario, tomando cualquier otro número natural k', en T se podría demostrar

$$\neg\bigwedge x \in \mathbb{N}(\phi(x) \leftrightarrow x = 0^{(k')}).$$

pero ϕ es (la traducción a $\mathcal{L}$ de) una fórmula de tipo Σ_1 de $\mathcal{L}_a$, por lo que si fuera verdadera en su interpretación natural, sería demostrable en Q, luego en T. Así pues, la fórmula ϕ no la cumple realmente ningún número natural, pero es consistente que cualquiera de ellos sea el único que la cumple. Más explícitamente, si $\phi(x) \equiv \bigvee u\, \alpha(x, u)$, donde α es de tipo Δ_0, tenemos que la fórmula $\alpha(x, u)$ es falsa en su interpretación natural, pero, para cada número natural k, es consistente que exista un número natural u tal que $\alpha(0^{(k)}, u)$. Dicho u se interpretará necesariamente como un número natural no estándar en cualquier modelo en el que $\phi(0^{(k)})$ sea verdadera. Podemos refinar aún más la conclusión:

Teorema 9.13 *Si T es una teoría semirrecursiva consistente que interpreta a* Q, *existe una fórmula*[5] *$\psi(x)$ de tipo Σ_1 tal que, si representamos $+\psi \equiv \psi$ y $-\psi \equiv \neg\psi$, entonces la teoría que resulta de añadir a T los axiomas*

$$\pm\psi(0), \quad \pm\psi(1), \quad \pm\psi(2), \quad \pm\psi(3), \quad \ldots$$

es consistente, para cualquier elección de los signos.

Demostración: Sea $\phi(x)$ la fórmula dada por el teorema anterior y sea

$$\psi(x) \equiv \bigvee s \in \mathbb{N}(\phi(s) \wedge x < \ell(s) \wedge s_x = 1).$$

Claramente es (la traducción de) una fórmula Σ_1 de $\mathcal{L}_a$. Puesto que una hipotética prueba de una contradicción usaría sólo un número finito de axiomas, basta probar que la teoría que resulta de añadir los axiomas $\pm\psi(0^{(0)}), \ldots, \pm\psi(0^{(n)})$ es consistente, para cada número natural n.

[5] Las fórmulas con esta propiedad se llaman fórmulas *flexibles*.

Tomamos, pues, un s tal que $\ell(s) = n$ y $\bigwedge i < n(s_i = 0 \vee s_i = 1)$ y consideremos la teoría T' dada por

$$T + \bigwedge x(\phi(x) \leftrightarrow x = \ulcorner s \urcorner).$$

Por el teorema anterior sabemos que T' es consistente, y es claro entonces que la s que aparece en la definición de ψ es necesariamente $\ulcorner s \urcorner$, es decir, que

$$\underset{T'}{\vdash} (\psi(x) \leftrightarrow x < 0^{(n)} \wedge \ulcorner s \urcorner_x = 1)$$

Así, para cada $i < n$, puesto que $\underset{T'}{\vdash} \ulcorner s \urcorner_{0^{(i)}} = 0^{(s(i))}$, es claro que $\underset{T'}{\vdash} \psi(0^{(i)})$ o $\underset{T'}{\vdash} \neg\psi(0^{(i)})$ según si $s(i)$ es cero o uno. Si la correspondiente extensión de T con estas sentencias fuera contradictoria, también lo sería T'. ∎

Así pues, aplicando el teorema a AP, concluimos que existe una fórmula $\psi(x)$ de $\mathcal{L}_a$ de tipo Σ_1 que en realidad no la cumple ningún número natural, pero los axiomas de Peano no permiten probarlo, sino que es consistente suponer que los números que queramos la cumplen y que cualesquiera otros no la cumplen. Y esto no es una deficiencia particular de los axiomas de Peano, sino que lo mismo sucede (cambiando la formula ψ por otra adecuada) para toda teoría axiomática que cumpla los requisitos mínimos usuales.

Cada extensión de AP con una determinación particular de una fórmula flexible nos da un modelo diferente de AP, en el sentido de que dos cualesquiera de ellos difieren al menos en que uno satisface una sentencia que el otro no satisface.

Observemos que la única construcción que conocemos de estos modelos es la que hemos usado en la demostración del teorema 7.37 (en LKD_0), que determina un modelo M a partir de un camino en un cierto árbol, cuya existencia viene garantizada por el lema de König débil, pero no tenemos un criterio explícito para determinar sus elementos. Terminamos esta sección viendo que esto no es casual. Vamos a demostrar que la suma y el producto en un modelo no estándar de AP no pueden ser funciones recursivas, lo que significa que toda construcción de un modelo no estándar de AP debe incluir un ingrediente "no constructivo" que impida operar explícitamente con la suma y el producto.

Para precisar esta idea veamos primero algunas consideraciones generales sobre modelos de AP:

De acuerdo con la definición 7.29, un modelo M de $\mathcal{L}_a$ está determinado por un universo M_0, al que llamamos simplemente M, en el que tenemos determinado un objeto $0_M = M_1(0)$, unas funciones

$$M_2(S) : M \longrightarrow M, \quad \oplus : M \times M \longrightarrow M, \quad \otimes : M \times M \longrightarrow M$$

y una relación $\leq_M$, de modo que podemos definir una función M_3 que a su vez determina la función $M(t)[v]$ y la relación $M \vDash \alpha[v]$ que cumplen las condiciones obvias de la definición de denotación y satisfacción.

De acuerdo con lo visto en la sección 7.5, llamaremos *números naturales estándar* de M a los objetos de la forma $0_M^{(n)} = M(0^{(n)})$, para cada número natural n.

Si M es un modelo de IA, el hecho de que las fórmulas del teorema 4.11 tengan que ser verdaderas en M se traduce en que la relación $\leq_M$ es una relación de orden total en M, y el teorema 6.2 nos da que la aplicación $\mathbb{N} \longrightarrow M$ dada por $n \mapsto 0_M^{(n)}$ es inyectiva y conserva el orden, es decir:

$$\bigwedge mn(m \leq n \leftrightarrow 0_M^{(m)} \leq 0_M^{(n)}).$$

Más aún, como M cumple la primera sentencia de 6.4, concluimos que

$$\bigwedge n \bigwedge a \in M(a \leq_M 0_M^{(n)} \to \bigvee m \leq n\, a = 0_M^{(m)}),$$

es decir, que todo número natural de M menor o igual que un número estándar es estándar o, dicho de otro modo, que cualquier número natural no estándar es mayor que cualquier número estándar. Por ello a los números naturales no estándar de un modelo se los llama también números naturales infinitos.

En particular vemos que el universo de M es necesariamente infinito, luego el teorema 7.23 nos da una biyección $\pi : \mathbb{N} \longrightarrow M$, a través de la cual podemos construir un modelo $\tilde{M}$ de $\mathcal{L}_a$ cuyo universo sea $\mathbb{N}$ y de modo que, para todo semitérmino t, toda semifórmula α y toda valoración v en $\tilde{M}$ definida sobre sus variables libres, se cumple

$$\tilde{M}(t)[v] = M(t)[\pi^{-1} \circ v], \qquad \tilde{M} \vDash \alpha[v] \leftrightarrow M \vDash \alpha[\pi^{-1} \circ v].$$

(De hecho, esto es esencialmente la definición de $\tilde{M}(t)[v]$ y $\tilde{M} \vDash \alpha[v]$.)

De este modo, $\tilde{M}$ es un modelo equivalente a M a todos los efectos, por lo que no perdemos generalidad si suponemos que el universo de M es todo el conjunto $\mathbb{N}$ de los números naturales. No obstante, seguiremos escribiendo M cuando pensemos en $\mathbb{N}$ como el universo de M.

Observemos que, si M es un modelo de IA, los teoremas 4.10 y 4.11 implican que las funciones $\oplus$ y $\otimes$ satisfacen las propiedades obvias, como $a \oplus b = b \oplus a$, etc. También tenemos que la función $M_1(S)$ que interpreta al funtor sucesor es simplemente $M_1(S)(a) = a \oplus 1_M$, por lo que no necesitamos trabajar con ella explícitamente.

Teorema 9.14 *Si M es un modelo de* Q *y $t(x_1, \ldots, x_n)$ es un término de $\mathcal{L}_a$, entonces, para todos los números naturales $a_1, \ldots, a_n$,*

$$M(t)[0_M^{(a_1)}, \ldots, 0_M^{(a_n)}] = 0_M^{(d)}, \quad \textit{con} \quad d = \mathrm{Dn}(t; a_1, \ldots, a_n),$$

donde el miembro hay que entenderlo como $\mathrm{Dn}(t, v)$*, para la valoración v dada por $v(x_i) = a_i$.*

DEMOSTRACIÓN: Se prueba, más en general, para semitérminos t, por inducción sobre t. La prueba es trivial, pero hay que comprobar que la fórmula a la que aplicamos el principio de inducción es Π^0_1 (requiere cuantificar sobre una valoración v). ∎

A su vez:

Teorema 9.15 *Si M es un modelo de* Q *y $\alpha(x_1,\dots,x_n)$ es una fórmula de $\mathcal{L}_a$ de tipo Δ_0, entonces*

$$M \vDash \alpha[0_M^{(a_1)},\dots,0_M^{(a_n)}] \leftrightarrow \mathbb{N} \vDash_0 \alpha[a_1,\dots,a_n].$$

DEMOSTRACIÓN: Se prueba para semifórmulas por inducción sobre α como en el teorema anterior. Los únicos casos no triviales son los correspondientes a los cuantificadores. Ambos son análogos, así que consideramos el del cuantificador universal, de modo que

$$\alpha \equiv \bigwedge u \le t(x_1,\dots,x_n)\, \beta(u,x_1,\dots,x_n).$$

Entonces $M \vDash \alpha[0_M^{(a_1)},\dots,0_M^{(a_n)}]$ si y sólo si para todo $d \in M$ tal que

$$d \le_M M(t)[0_M^{(a_1)},\dots,0_M^{(a_n)}],$$

se cumple $M \vDash \beta[d, 0_M^{(a_1)},\dots,0_M^{(a_n)}]$.

Por el teorema anterior, llamando $k = \mathrm{Dn}(t,a_1,\dots,a_n)$, la condición sobre d es que sea $d \le_M 0_M^{(k)}$, pero, por el teorema 6.4, los $d \in M$ que cumplen esto son los de la forma $d = 0_M^{(a)}$, con $a \le k$. Usando además la hipótesis de inducción, tenemos que se cumple $M \vDash \alpha[0_M^{(a_1)},\dots,0_M^{(a_n)}]$ si y sólo si, para todo $a \le \mathrm{Dn}(t,a_1,\dots,a_n)$, se cumple $\mathbb{N} \vDash_0 \beta[a,a_1,\dots,a_n]$, pero esto equivale a $\mathbb{N} \vDash_0 \bigwedge u \le t\, \beta(u,x_1,\dots,x_n)[a_1,\dots,a_n]$, es decir, a $\mathbb{N} \vDash_0 \alpha[a_1,\dots,a_n]$. ∎

En particular vemos que todos los modelos de Q cumplen las mismas sentencias de tipo Δ_0. Para fórmulas de tipo Σ_1 o Π_1 tenemos las consecuencias siguientes:

Teorema 9.16 *Sea M un modelo de* Q*. Si $\alpha(x_1,\dots,x_n)$ es una fórmula de $\mathcal{L}_a$ de tipo Σ_1, entonces*

$$\mathbb{N} \vDash_{\Sigma_1} \alpha[a_1,\dots,a_n] \to M \vDash \alpha[0_M^{a_1},\dots,0_M^{a_n}].$$

Si α es de tipo Π_1, entonces

$$M \vDash \alpha[0_M^{a_1},\dots,0_M^{a_n}] \to \mathbb{N} \vDash_{\Pi_1} \alpha[a_1,\dots,a_n].$$

DEMOSTRACIÓN: Si α es Σ_1, entonces $\alpha \equiv \bigvee u\, \beta(u,x_1,\dots,x_n)$, para cierta fórmula $\beta(x_0,\dots x_n)$ de tipo Δ_0. Si $\mathbb{N} \vDash_{\Sigma_1} \alpha[a_1,\dots,a_n]$, existe un a_0 tal que $\mathbb{N} \vDash_0 \beta[a_0,\dots,a_n]$, luego por el teorema anterior $M \vDash \beta[0_M^{(a_0)},\dots,0_M^{(a_n)}]$, y esto implica que $M \vDash \alpha[0_M^{(a_1)},\dots,0_M^{(a_n)}]$.

En el segundo caso $\alpha \equiv \bigwedge u\, \beta(u, x_1, \ldots, x_n)$ y, si $M \vDash \alpha[0_M^{a_1}, \ldots, 0_M^{a_n}]$, tenemos que para todo $d \in M$ se cumple $M \vDash \beta[m, 0_M^{a_1}, \ldots, 0_M^{a_n}]$, luego, en particular, para todo a_0 se cumple $M \vDash \beta[0_M^{a_0}, \ldots, 0_M^{a_n}]$. El teorema anterior nos da que $\mathbb{N} \vDash_0 \beta[a_0, \ldots, a_n]$, y esto implica que $\mathbb{N} \vDash_{\Pi_1} \alpha[a_1, \ldots, a_n]$. ∎

Si α es una sentencia de $\mathcal{L}_a$ de tipo Π_1, según 6.15, en $\mathrm{I}\Sigma_1$ se demuestra

$$\alpha \leftrightarrow \mathbb{N} \vDash_{\Pi_1} \ulcorner\alpha\urcorner,$$

luego, si M es un modelo de $\mathrm{I}\Sigma_1$, esta sentencia es verdadera en M, por lo que $M \vDash \alpha$ es equivalente a $M \vDash (\mathbb{N} \vDash_{\Pi_1} \ulcorner\alpha\urcorner)$.

Consideremos ahora la fórmula $\phi_0(\alpha)$ de $\mathcal{L}_a^0$ que afirma que α es una sentencia de $\mathcal{L}_a$ de tipo Π_1 y sea $\phi \equiv \ulcorner\phi_0\urcorner \in \mathrm{Form}(\mathcal{L}_a)$ su formalización. La fórmula $\phi_0(\alpha)$ es de tipo Δ_1 en $\mathrm{I}\Sigma_1$, es decir, es Σ_1 y existe una fórmula $\psi_0(\alpha)$ de tipo Π_1 de modo que $\bigwedge u(\phi_0(u) \leftrightarrow \psi_0(u))$, luego, si $\psi \equiv \ulcorner\psi_0\urcorner$, se cumple

$$\underset{\mathrm{I}\Sigma_1}{\vdash} \bigwedge u(\phi(u) \leftrightarrow \psi(u)).$$

Por consiguiente, si M es un modelo de $\mathrm{I}\Sigma_1$, para todo número natural a, se cumple

$$M \vDash \phi[a] \leftrightarrow M \vDash \psi[a].$$

Por el teorema anterior, para todo número natural α,

$$\mathbb{N} \vDash_{\Sigma_1} \phi[\alpha] \to M \vDash \phi[0_M^{(\alpha)}], \qquad M \vDash \psi[0_M^{(\alpha)}] \to \mathbb{N} \vDash_{\Pi_1} \alpha.$$

Por otro lado, 6.16 nos da que

$$\bigwedge \alpha(\phi_0(\alpha) \leftrightarrow \mathbb{N} \vDash_{\Sigma_1} \phi[\ulcorner\alpha\urcorner]), \qquad \bigwedge \alpha(\psi_0(\alpha) \leftrightarrow \mathbb{N} \vDash_{\Pi_1} \psi[\ulcorner\alpha\urcorner]).$$

Combinando estas implicaciones vemos que

$$\phi_0(\alpha) \to \mathbb{N} \vDash_{\Sigma_1} [\ulcorner\alpha\urcorner] \to M \vDash \phi[0_M^{(\alpha)}] \to$$

$$M \vDash \psi[0_M^{(\alpha)}] \to \mathbb{N} \vDash_{\Pi_1} \psi[\ulcorner\alpha\urcorner] \to \psi_0(\alpha) \to \phi_0(\alpha).$$

En resumen, un número natural α es una sentencia Π_1 de $\mathcal{L}_a$ si y sólo si $\mathbb{N} \vDash_{\Sigma_1} \phi[\ulcorner\alpha\urcorner]$ si y sólo si $M \vDash \phi[0_M^{(\alpha)}]$.

Ahora consideramos la fórmula $\psi_0(n, m) \equiv p_n \mid m$, donde p_n es el primo n-simo. Es también una fórmula de $\mathcal{L}_a^0$ de tipo Δ_1 en $\mathrm{I}\Sigma_1$, por lo que, si llamamos $\psi \equiv \ulcorner\psi_0\urcorner$, el mismo razonamiento precedente nos da que, para todos los números naturales n y m, se cumple $p_n \mid m$ si y sólo si $\mathbb{N} \vDash_{\Sigma_1} \psi[0^{(n)}, 0^{(m)}]$, si y sólo si $M \vDash \psi[0_M^{(n)}, 0_M^{(m)}]$. Usamos estos hechos en la prueba del teorema siguiente:

Teorema 9.17 *Sea M un modelo no estándar de* AP *y sea $\psi(n, m)$ la fórmula de $\mathcal{L}_a$ que expresa que el primo n-simo divide a m. Entonces existe un número natural no estándar $c \in M$ tal que el conjunto*

$$S = \{i \mid M \vDash \psi[0_M^{(i)}, c]\}$$

no es recursivo.

DEMOSTRACIÓN: Sea $\phi(\alpha)$ la fórmula que hemos considerado en la discusión previa al enunciado. Definimos

$$\chi(y) \equiv \bigvee x \bigwedge \alpha < y(\psi(\alpha, x) \leftrightarrow \phi(\alpha) \wedge \mathbb{N} \vDash_{\Pi_1} \alpha).$$

Observemos que todo número natural n cumple $M \vDash \chi[0_M^{(n)}]$. En efecto, esto significa que existe un $d \in M$ tal que para todo $e \in M$ que cumpla $e <_M d$, se cumple $M \vDash \psi[e, d]$ si y sólo si $M \vDash \phi[e]$ y $M \vDash (\mathbb{N} \vDash_{\Pi_1} x)[e]$.

Para probar que esto es así consideramos el producto m de todos los primos p_α tales que $\alpha < n$ es una sentencia de $\mathcal{L}_a$ de tipo Π_1 tal que $M \vDash \alpha$ y tomamos $d = 0_M^{(m)}$. Así, si $e \in M$ cumple $e <_M 0_M^{(n)}$, existe un número $\alpha < n$ tal que $e = 0_M^{(\alpha)}$. Entonces, $M \vDash \psi[e, d]$ equivale a $M \vDash \psi[0_M^{(\alpha)}, 0_M^{(m)}]$ y, según las observaciones previas al teorema, esto equivale a $p_\alpha \mid m$, lo que, por la elección de m, equivale a que α sea una sentencia de $\mathcal{L}_a$ de tipo Π_1 tal que $M \vDash \alpha$. De nuevo por la discusión previa al enunciado, esto equivale a que $M \vDash \phi[0_M^{(\alpha)}]$ y $M \vDash (\mathbb{N} \vDash_{\Pi_1} \ulcorner\alpha\urcorner)$, que a su vez equivale a[6] $M \vDash \phi[e]$ y $M \vDash (\mathbb{N} \vDash_{\Pi_1} x)[e]$.

De aquí se sigue que existe un $d \in M$ no estándar tal que $M \vDash \chi[d]$. En efecto, en caso contrario, los elementos de M que cumplen $M \vDash \chi[d]$ serían exactamente los números estándar, luego tendríamos que[7]

$$M \vDash \chi(0) \wedge \bigwedge u(\chi(u) \to \chi(u+1)).$$

Como M satisface el principio de inducción de AP, debería cumplirse también $M \vDash \bigwedge u\, \chi(u)$, pero esto significa que todos los elementos de M son estándar, en contra de lo supuesto.

Así pues, fijado un número no estándar $d \in M$ que cumpla $M \vDash \chi[d]$, por definición de χ existe un $c \in M$ tal que, para todo $\alpha \in M$ tal que $\alpha <_M d$, se cumple

$$M \vDash \psi[\alpha, c] \leftrightarrow M \vDash \phi[\alpha] \wedge M \vDash (\mathbb{N} \vDash_{\Pi_1} x)[\alpha].$$

Por consiguiente, $\alpha \in S$ si y sólo si $M \vDash \psi[0_M^{(\alpha)}, c]$ y, como $0_M^{(\alpha)} <_M d$, puesto que d es no estándar, esto equivale a

$$M \vDash \phi[0_M^{(\alpha)}] \wedge M \vDash (\mathbb{N} \vDash_{\Pi_1} \ulcorner\alpha\urcorner),$$

que a su vez equivale a que α es una sentencia Π_1 de $\mathcal{L}_a$ tal que $M \vDash \alpha$. En resumen: S es el conjunto de todas las sentencias Π_1 de $\mathcal{L}_a$ verdaderas en M, que no es recursivo por el teorema 9.6. ∎

Con esto ya podemos demostrar:

---

[6] Notemos que

$$M \vDash (\mathbb{N} \vDash_{\Pi_1} \ulcorner\alpha\urcorner) \equiv M \vDash (\mathbb{N} \vDash_{\Pi_1} 0^{(\alpha)}) \leftrightarrow M \vDash (\mathbb{N} \vDash_{\Pi_1} x)[0_M^{(\alpha)}] \leftrightarrow M \vDash (\mathbb{N} \vDash_{\Pi_1} x)[e].$$

[7] La fórmula χ es Σ_2, luego en realidad basta con que M sea un modelo de $\mathrm{I}\Sigma_1$.

Teorema 9.18 (Tennenbaum) *Si un modelo no estándar M de* AP *tiene universo $\mathbb{N}$, las funciones que interpretan en M los funtores suma y producto de $\mathcal{L}_a$ no son recursivas.*

DEMOSTRACIÓN: Sea $c \in M$ un número no estándar según el teorema anterior, de modo que el conjunto

$$S = \{i \mid M \vDash \psi[0_M^{(i)}, c]\}$$

no es recursivo. Vamos a suponer que la suma $\oplus$ en M es recursiva y probaremos que el conjunto S también lo es, con lo que tendremos una contradicción.

Si $\oplus$ es recursiva, también lo es la función dada por

$$p(x, 0) = 0_M, \quad p(x, n+1) = p(x, n) \oplus x.$$

Una simple inducción prueba que

$$0_M^{(n)} \otimes x = p(x, n).$$

En particular $p(1_M, n) = 0_M^{(n)}$.

Como M satisface el teorema de la división euclídea, fijado i, existen unos únicos $k, r' \in M$ tales que

$$c = 0_M^{(p_i)} \otimes k \oplus r', \qquad r' <_M 0_M^{(p_i)}.$$

La desigualdad implica que r' es un número estándar, es decir, que existe un $r < p_i$ tal que $r' = 0_M^{(r)} = p(1_M, r)$. Por lo tanto, tenemos que existen unos únicos k y $r < p_i$ tales que

$$c = p(k, p_i) \oplus p(1_M, r).$$

Esto implica que la función dada por

$$f(i) = \mu n(c = p(n_0, p_i) \oplus p(1_M, n_1) \wedge n_1 < p_i)$$

es recursiva, al igual que la función $R(i) = f(i)_1$, es decir, la función que da el r tal que $0_M^{(r)}$ es el resto de la división euclídea de c entre $0_M^{(p_i)}$.

Como la fórmula $\phi(m, n) \equiv m = p_n$ es Σ_1, el teorema 9.16 nos da que si $j = p_i$ es el primo i-ésimo, entonces $\mathbb{N} \vDash \phi[j, i]$, luego $M \vDash \phi[0_M^{(j)}, 0_M^{(i)}]$, es decir, que $0_M^{(p_i)} = 0_M^{(j)}$ es el primo $0_M^{(i)}$-ésimo en M.

Finalmente, $i \in S$ equivale a $M \vDash \psi[0_M^{(i)}, c]$, que a su vez significa que el primo $0_M^{(i)}$-ésimo de M (es decir, $0_M^{(p_i)}$) divide a c, y que divida a c equivale a que el resto de la división euclídea sea 0_M o, equivalentemente, a que $R(i) = 0$. Así pues, $i \in S$ si y sólo si $R(i) = 0$, luego la función característica de S es $1 \dot{-} (1 \dot{-} R)$, luego es recursiva y S es un conjunto recursivo, que es la contradicción que perseguíamos.

Supongamos ahora que $\otimes$ es una función recursiva (pero no que lo sea $\oplus$) y definamos igualmente la función recursiva

$$p'(x,n) = 1_M, \quad p'(x,n+1) = p'(x,n) \otimes x.$$

En AP se demuestra que

$$\bigwedge xy \overset{1}{\bigvee} z\, x^y = z,$$

luego podemos definir la función $\exp : M^2 \longrightarrow M$ que, para cada $a, b \in M$, cumple

$$M \vDash (x^y = z)[a, b, \exp(a,b)],$$

y una simple inducción prueba que $\exp(x, 0_M^n) = p'(x,n)$.

Por otro lado, en AP se demuestra que la relación $x = yc + r$ es equivalente a $2^x = (2^c)^y \cdot 2^r$ y, más aún, que el resto r de la división euclídea de x entre y es el único número $r < y$ tal que existe un z que cumple $2^x = z^y \cdot 2^r$. Por lo tanto, si llamamos $d = \exp_2(c)$, tenemos que el resto de la división euclídea de c entre $0_M^{(p_i)}$ es el único $r' < 0_M^{(p_i)}$ para el que existe un $t \in M$ que cumple

$$d = \exp(t, 0_M^{(p_i)}) \otimes \exp(2_M, r'),$$

pero tiene que ser $r' = 0_M^{(r)}$, con $r < p_i$, con lo que

$$d = \exp(t, 0_M^{(p_i)}) \otimes \exp(2_M, 0_M^{(r)}) = p'(t, p_i) \otimes p'(2_M, r),$$

y el resto de la división euclídea de c entre $0_M^{(p_i)}$ es 0_M si y sólo si el único $r < p_i$ que cumple

$$d = p'(t, p_i) \otimes p'(2_M, r)$$

para cierto $t \in M$ es $r = 0$. Ahora podemos definir como antes la función recursiva

$$f'(i) = \mu n(d = p'(n_0, p_i) \otimes p'(2_M, n_1) \wedge n_1 < p_i),$$

que a su vez nos da una definición alternativa $R(i) = f(i)_1$ de la misma función R que antes habíamos definido en términos de $\oplus$, y llegamos igualmente a que es recursiva, con lo que el conjunto S también lo es, y esta contradicción prueba que el producto $\otimes$ no puede ser recursivo. ∎

Así pues, no es posible definir explícitamente una suma y un producto en $\mathbb{N}$ (de forma que podamos calcularlas en la práctica) con las que $\mathbb{N}$ se convierta en un modelo no estándar de AP.

Índice de Materias

ACA_0, 330
ACR_0, 333
antiimagen, 92
aplicación, 92
    inyectiva, suprayectiva, biyectiva, 93
árbol, 354
aritmética
    con inducción abierta / Δ_0, 193
    de Peano
        de primer orden, 202
        de segundo orden, 330
    recursiva primitiva, 27
        de primer orden, 186
ARP, 27
ARP_0^2, 335
axioma, 148
    de ARP, 27
    de especificación, 247
    de extensionalidad, 247
    de recolección, 252
    de regularidad, 252
axiomas de Peano, 202
    en ARP, 45

bien fundado (conjunto), 232

cardinal, 94, 269
clase, 230
    propia, 231
clausura transitiva, 90
código, 392
composición, 8, 388
    de funciones, 92
    parcial, 411
conexo (conjunto), 232
consecuencia, 148
    inmediata, 148
    lógica, 23
coprimos, 109

deducción, 148
    en ARP, 27
definición de un funtor de ARP, 19
demostrablemente recursiva (función), 419
demostración, 148
    en ARP, 27
denotación
    de un funtor en ARP, 9
    de un término en ARP, 16
desarrollo decimal, 84
designador, 143
divisibilidad, 108
dominio, 91

enteros (números), 101
equipotencia, 93
especificación, 256
expresión, 142
    abierta/cerrada, 143
extensión
    conservativa, 178
    intrasdendente, 178
    predicativa, 325

flexible (fórmula), 446
forma prenexa, 175
fórmula, 142
    de ARP, 17
funtor de ARP, 7

Gödel (sentencia de), 432

Hilbert-Bernays (condiciones de), 318

imagen (por una aplicación), 92
interpretación, 241

Kleene (jerarquía de), 200, 209

Lema
  de König débil, 358
  de Lindenbaum, 358
lenguaje formal
  de la aritmética
    de primer orden, 192
    de segundo orden, 329
  de la teoría de conjuntos, 227
  de primer orden, 132
  de segundo orden, 322
    reducido, 323
LKD_0, 358
Lévy (jerarquía de), 238

minimización, 389
  parcial, 412

numeral, 284
  de ARP, 4
número natural, 237

ordinal, 232

par ordenado
  en ARP, 74
  en B, 229
  en IA, 198
pertenencia, 87
potencia (conjunto), 93
prefijo, 175
premisa, 148
primo, 109
proyección (en ARP), 5

RA_0, 383
racionales (números), 107
rango, 91
recolección, 203, 255
recursión, 8, 259, 389
  aritmética, 383
  completa, 80
  parcial, 412
recursiva (función), 396
  parcial, 413
recursiva (teoría), 317
recursivo (conjunto), 423
reducción al absurdo, 157
  en ARP, 59
reducción, 255
reemplazo, 257
reflexión, 254
reglas de inferencia, 148
  de ARP, 26
relatores estructurales, 322
Robinson (aritmética de), 284

satisfacción, 140
  en ARP, 18
semifórmula, 139
  estructurada, 323
semirrecursiva (teoría), 317
semitérmino, 139
  estructurado, 323
sentencia, 143
signo eventual/obligatorio, 133
sistema deductivo formal, 148
sucesiones finitas, 77
sumas finitas, 83, 98
sustitución, 145
  en ARP, 21

Teorema
  de Σ_1-completitud de Q, 305
  de Church, 436
  de compacidad, 366
  de completitud semántica, 360, 365
  de corrección, 352
    de ARP, 28
  de Craig, 317
  de deducción, 154
    en ARP, 52
  de incompletitud de Gödel
    primero, 432
    segundo, 438
    versión de Rosser, 433
  de Löb, 441
  de reflexión, 311

de Tarski, 442
de Tennenbaum, 452
teorema, 149
de ARP, 27
lógico, 152
teoría
axiomática, 177
de segundo orden, 324
básica de conjuntos, 227
de Kripke-Platek, 252
de Zermelo, 247
de Zermelo-Fraenkel, 267
término, 142
de ARP, 14
transitivo (conjunto), 90, 232

valoración, 140
en ARP, 16
variable
de ARP, 14
libre/ligada, 142
verdadero/falso, 145
en ARP, 18